MILTON BORSATO

UMA PLATAFORMA DE SUPORTE AO GERENCIAMENTO DO DESENVOLVIMENTO RÁPIDO DE PRODUTOS TECNOLÓGICOS ATRAVÉS DA ENGENHARIA SIMULTÂNEA

Tese apresentada ao Programa de Pós-Graduação em Engenharia de Produção, Centro de Tecnologia e Ciências, Universidade Federal de Santa Catarina, como requisito parcial à obtenção do título de Doutor em Engenharia.

Orientador: Prof. Rolf Hermann Erdmann, Dr.

FLORIANÓPOLIS

2003

Ficha Catalográfica

B738p Borsato, Milton

Uma plataforma de suporte ao gerenciamento do desenvolvimento rápido de produtos tecnológicos através da engenharia simultânea / Milton Borsato . - Curitiba : [s.n.], 2003.

xv, 227 f. : il. ; 30 cm

Orientador : Prof. Dr. Rolf Hermann Erdmann

Tese (Doutorado) – UFSC. Programa de Pós-Graduação em Engenharia de Produção. Centro de Tecnologia e Ciências. Florianópolis, 2003.

Bibliografia : f. 150-56

1. Projetos de engenharia. 2. Engenharia de produção. 3. Engenharia concorrente. 4. Produtos novos. 5. Administração de produto. 6. Projeto auxiliado por computador (Engenharia). 7. Tecnologia da Informação. 8. Sistema CAD/CAM/CAE. 9. Java (Linguagem de programação de computador). 10. Sistema métrico. I. Erdmann, Rolf Hermann, orient. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Produção. Centro de Tecnologia e Ciências. III. Título.

CDD : 620.0042
CDU : 658.512.2

MILTON BORSATO

UMA PLATAFORMA DE SUPORTE AO GERENCIAMENTO DO DESENVOLVIMENTO RÁPIDO DE PRODUTOS TECNOLÓGICOS ATRAVÉS DA ENGENHARIA SIMULTÂNEA

Esta tese foi julgada e aprovada para a obtenção do grau de **Doutor em Engenharia de Produção** no **Programa de Pós-Graduação em Engenharia de Produção** da Universidade Federal de Santa Catarina.

Florianópolis, 18 de junho de 2003.

Prof. Edson Pacheco Paladini, Dr.
Coordenador do Programa

BANCA EXAMINADORA

Prof. Rolf Hermann Erdmann, Dr.
Universidade Federal de Santa Catarina
Orientador

Prof. Fernando A. Forcellini, Dr.
Universidade Federal de Santa Catarina

Prof. André Ogliari, Dr.
Universidade Federal de Santa Catarina

Prof. Kazuo Hatakeyama, Ph.D.
University of Wales

Prof. Carlos Cziulik, Ph.D.
University of Surrey

Prof. Nelson Casarotto Filho, Dr.
Universidade Federal de Santa Catarina

DEDICATÓRIA

Este trabalho é dedicado a Paola, Giovanna e Karin, pela inspiração, compreensão e apoio nos momentos em que mais precisei.

AGRADECIMENTOS

Esta tese de doutorado é fruto de um esforço que começou no final de 1994 com a criação do Núcleo de Pesquisa em Engenharia Simultânea (NuPES) através de uma parceria entre o Centro Federal de Educação Tecnológica do Paraná (posteriormente FUNCEFET-PR) e a Equitel Telecomunicações (posteriormente Siemens LTDA.), viabilizada pela Lei 8.248/91, conhecida como a Lei de Informática. Agradeço a todos que colaboraram no CEFET-PR e nas empresas parceiras para viabilizar esta iniciativa.

Através de mecanismos legais de renúncia fiscal, vários grupos de pesquisadores brasileiros tiveram a oportunidade de trabalhar por uma indústria nacional mais competitiva, ao cooperarem com engenheiros e projetistas da indústria para o desenvolvimento de novos produtos e tecnologias na área de informática e telecomunicações. Entre eles, um grupo de jovens pesquisadores acreditou nesta causa, e transformou uma idéia num núcleo que desponta como referência nacional em suas linhas de atuação. Muitos dos conceitos aqui apresentados foram assimilados através da biblioteca do núcleo e das várias participações em cursos, congressos e eventos, nacionais e internacionais, dos quais tive a oportunidade de participar.

Portanto agradeço a todos os colegas professores que já fizeram e que atualmente fazem parte deste grupo, pelo incessante apoio e compreensão. Em especial, agradeço ao Prof. Gilson Yukio Sato, que de forma incansável supriu as necessidades de liderança no NuPES quando não pude estar presente.

Aos companheiros gerentes, engenheiros e projetistas da Siemens LTDA., agradeço pela sempre pronta cooperação, não só para o levantamento de informações técnicas para a execução do trabalho, mas também por terem demonstrado interesse ao apoiar uma parte significativa deste projeto de janeiro a dezembro de 2001, através da Lei de Informática.

Não obstante, agradeço também aos Professores Alfredo Vrubel, Richard Duarte Ribeiro, Gilda Maria Souza Friedlaender, Luiz Nacamura Jr., Mauricio Alves Mendes e Marcelo Mikosz Gonçalves pelo apoio na liberação das atividades de ensino no Departamento Acadêmico de Informática (DAINF) da Unidade de Curitiba, durante o período de doutoramento.

Aos professores do DAINF Leandro Batista de Almeida e Caio Nakashima, agradeço por compartilharem de seu conhecimento técnico durante processo de implementação do sistema PSES. Da mesma forma, agradeço aos servidores Douglas Irineu e Carlos Kazuhiko Hara, pelo apoio na área de administração de sistemas e redes.

Aos acadêmicos de Engenharia Industrial Elétrica do CEFET-PR, e ex-estagiários do NuPES, Mariana Pucci da Silva e Rafael Machado Bardal, que aceitaram o desafio e participaram de forma dinâmica em todas as fases deste trabalho, agradeço pela confiança, mesmo nos momentos mais difíceis.

Ao Prof. Dr. Rolf Hermann Erdmann, orientador neste trabalho, agradeço pela confiança incondicional depositada, incentivo e críticas, no sentido de nortear os trabalhos sob o ponto de vista acadêmico.

Por fim, agradeço a meus familiares e amigos pela compreensão e incentivo, para que mais este desfio fosse superado.

EPÍGRAFE

As idéias da Engenharia Simultânea existem há muito tempo, mas não foram colocadas em prática tanto porque métodos mais antigos pareciam mais fáceis, quanto porque o sistema educacional não defendeu suficientemente que uma mudança deveria ser feita. Educadores e pesquisadores têm a obrigação de assegurar que isto não ocorra novamente.

Robert P. Smith

SUMÁRIO

LISTA DE ILUSTRAÇÕES

LISTA DE SIGLAS

BPR	*Business Process Reengineering*
CAD	*Computer-Aided Design*
CAE	*Computer-Aided Engineering*
CAM	*Computer-Aided Manufacturing*
CAPP	*Computer-Aided Process Planning*
CASE	*Computer-Aided Software Engineering*
CE	*Concurrent Engineering*
CGI	*Common Gateway Interface*
CSCW	*Computer-Supported Cooperative Work*
DER	Diagrama de Relacionamento de Entidades
DFA	*Design for Assembly*
DFM	*Design for Manufacturing*
DSM	*Design Structure Matrix*
DIP	Desenvolvimento Integrado de Produto
ERP	*Enterprise Resource Planning*
ES	Engenharia Simultânea
FTP	*File Transfer Protocol*
HTTP	*Hyper Text Transfer Protocol*
ICAM	*Integrated Computer-Aided Manufacturing*
IDA	*Institute of Defense Analysis*
IDE	*Integrated Development Environment*
IDEF	*ICAM Definition*
IDEF3	*ICAM Definition Method for Process Description Capture*
IGES	*Initial Graphics Exchange Specification*
INCOSE	*International Council on Systems Engineering*
ISO	*International Organization for Standardization*
ISPE	*International Society for Productivity Enhancement*
J2EE	*Java 2, Enterprise Edition*

JDBC	*Java Database Connectivity*
JSP	*Java Server Pages*
MIME	*Multipurpose Internet Mail Extensions*
OO	Orientação a Objetos
OSTN	*Object State Transition Network*
P&D	Pesquisa e Desenvolvimento
PDF	*Portable Document Format*
PDM	*Product Data Management*
PHP	*Hypertext Preprocessor*
PLM	*Product Lifecycle Management*
PSES	Plataforma de Suporte à Engenharia Simultânea
QFD	*Quality Function Deployment*
RPG	*Role Playing Games*
SOCE	*Society of Concurrent Engineering*
SQL	*Structured Query Language*
STEP	*Standard for the Exchange of Product Model Data*
UML	*Uniform Modeling Language*
XML	*Extensible Markup Language*
WWW	*World Wide Web*

RESUMO

BORSATO, Milton. **Uma plataforma de suporte ao gerenciamento do desenvolvimento rápido de produtos tecnológicos através da engenharia simultânea**. 2003. 228f. Tese (Doutorado em Engenharia de Produção) - Programa de Pós-Graduação em Engenharia de Produção, UFSC, Florianópolis.

O presente estudo propõe uma estrutura de organização de trabalho baseada nos preceitos da Engenharia Simultânea (ES). Os objetivos foram: (a) levantamento de cenários que contêm modelos de informação e processos para representar o desenvolvimento de produtos; (b) especificação e implementação de um sistema de informação, através do qual os preceitos da ES são aplicados em todas as suas abordagens e; (c) definição de um sistema de métricas que permite avaliar os trabalhos de uma equipe de desenvolvimento sob a ótica da ES. A pesquisa foi de natureza aplicada, utilizou uma abordagem qualitativa e teve caráter exploratório-descritivo quanto aos seus objetivos. Adotou ainda procedimentos técnicos que a classificam como do tipo bibliográfico, de levantamento, pesquisa-ação, participativa e experimental. Quanto aos procedimentos adotados, foram escolhidos dois cenários de desenvolvimento de produto, cujo trabalho de levantamento consistiu em registrar e avaliar entrevistas com engenheiros, projetistas e demais colaboradores de um equipe de desenvolvimento, apurar documentos de referência corporativa para o desenvolvimento de produto, e aplicar métodos reconhecidos, além de outras técnicas desenvolvidas especificamente para este fim. Futuros levantamentos de cenários poderão seguir os mesmos procedimentos. Em seguida, foi especificado um sistema de informações que combina funcionalidades de ferramentas de gestão de dados de projeto, gestão do trabalho colaborativo, gestão do fluxo de trabalho e gestão de recursos de projetos, entre outras. Os cenários levantados são utilizados com a finalidade de permitir a construção de um sistema que atua como agente facilitador para a prática da ES no desenvolvimento de produtos de teor tecnológico. Ao mesmo, tempo, novos cenários podem ser agregados ao sistema de forma a ampliar a sua aplicabilidade. Uma vez especificado o sistema, o banco de dados relacional foi definido através de um diagrama de relacionamento de entidades. As informações dos cenários levantados foram inicialmente implementadas no banco de dados através de *scripts* SQL, arquivos XML referentes a cronogramas mestres e páginas JSP contendo elementos de navegação em fluxo de trabalho utilizando *applets* Java. O sistema de informação PSES foi modelado em padrão UML e codificado em linguagem Java, utilizando-se técnicas de programação orientada a objeto para ambiente *web*. Foram projetadas dezenas de páginas JSP e Javabeans, que consultam uma mesma base relacional de dados através de um *pool* de conexões via JDBC. O usuário utiliza um navegador convencional para acessar o sistema via protocolo HTTP. No sistema PSES, as diversas funcionalidades estão agrupadas em três modos de uso: administração, gestão e operação. O modo de administração permite o cadastramento de novos usuários, empresas, setores, tipos de arquivos e repositórios a serem utilizados, entre outros. O modo de operação permite ao usuário colaborador de uma equipe de desenvolvimento consultar e modificar dados de projeto através de ações de *upload* e *download* de arquivos, conforme seu nível e área de responsabilidade no projeto, atualizar indicadores de progresso das tarefas sobre as quais responde e agendar sessões colaborativas síncronas e assíncronas. O modo de gestão permite ao gerente de projeto operar elementos de informação mestres, como requisitos do cliente e cronograma, configurar o mesmo para a criação de uma equipe, monitorar o fluxo do trabalho e o desempenho da equipe através de diagramas, utilizando como critério o sistema de métricas proposto.

Palavras chave: engenharia simultânea, desenvolvimento integrado de produtos, sistema de informação, desenvolvimento de produtos.

ABSTRACT

BORSATO, Milton. **A platform to support management of rapid technological product development through concurrent engineering**. 2003. 228f. Tese (Doutorado em Engenharia de Produção) - Programa de Pós-Graduação em Engenharia de Produção, UFSC, Florianópolis.

The present study proposes a framework based on Concurrent Engineering (CE) principles. The objectives of this work were: (a) to build scenarios that contain information as well as process models so that product development can be represented; (b) to specify and implement an information system, in which CE concepts are applied throughout all its different approaches and; (c) to define a metrics system that allows the evaluation of the work performed by a team under a CE point of view. This research has presented applied nature, exercised a qualitative approach and had an exploratory-descriptive character for its objectives. Yet, it has adopted technical procedures that classify it as of bibliographic type, of information gathering, action-research, participative and experimental. As for the adopted procedures, two product development scenarios have been selected, in which the building work consisted of registering and evaluating interviews with engineers, designers and other collaborates of a development team, analyzing corporate reference documentation for product development and, applying well-known methods, as well as others designed specifically for this purpose. Future scenario building will possibly follow similar procedures. Next, an information system was specified, combining several features such as those of product data management tools, groupware tools, project management tools, workflow tools, among others. The scenarios built previously were used to help implement a system that acts as a facilitator agent for the practice of CE in the development of products of extensively applied technology. Moreover, new scenarios can be added to the system in order to extend its applicability. Once the system was specified, a relational database has been modeled through an entity relation diagram. The information about the previously built scenarios were initially implemented onto the database through SQL scripts, master schedule files in XML format and JSP pages that contain workflow navigation elements in Java applets. The information system named PSES has been modeled in UML and coded in Java language, using the object oriented paradigm for the Web environment. Several JSP pages and Javabeans have been designed, which consult a unique relational database through a pool of connections via JDBC. The user navigates in a conventional browser to access the system via HTTP protocol. In the PSES system, several features have been grouped around three use modes: administration, management and operation. The administration mode allows the registration of new users, enterprises, departments, file types and repositories to be used, among other possibilities. The operation mode allows the user, member of a development team, to consult and modify project data by way of uploading and downloading actions, depending on his/her level and area of responsibility in the project, to update task progress indicators of which he/she is liable for and to schedule synchronous and asynchronous collaborative sessions. The management mode allows the project manager to operate master elements of information, like customer needs and schedule, to configure the system to accommodate a development team, to monitor workflow and team performance with graphics, utilizing criteria based on the metrics system proposed.

Keywords: concurrent engineering integrated product development, information system, product development.

1 INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

Os anos noventa trouxeram para as empresas do setor secundário brasileiro uma nova realidade de competição em âmbito mundial. O avanço das tecnologias de comunicação, a integração entre países através de blocos comerciais, a disseminação de empresas denominadas globais pelos cinco continentes e os diferentes custos de mão-de-obra e incentivos fiscais entre os países industrializados e em desenvolvimento levaram a uma situação em que qualquer redução significativa nos custos de produção pode alterar a distribuição das plantas industriais no mundo.

Entretanto, não só a redução dos custos de produção é relevante para a sobrevivência de empresas que concebem e comercializam produtos de cunho tecnológico. A qualidade e o atendimento às expectativas do cliente no tocante à funcionalidade, preço e inovação também exercem papel decisivo no posicionamento das empresas diante do mercado consumidor.

As condições dinâmicas do mercado e a competição internacional estão fazendo com que o parâmetro *time-to-market*[1] venha sendo reduzido drasticamente pelas empresas nos últimos anos (HUNDAL, 1998). Fundamentalmente, o mercado se comporta desta maneira em função das oportunidades para estabelecimento de unidades de manufatura geograficamente distribuídas, além das condições diferenciadas das economias e processos constantes de inovação tecnológica.

Por outro lado, as necessidades dos clientes evoluem constantemente. O mercado consumidor está cada vez mais sofisticado, pois uma vez que suas necessidades sejam satisfeitas num produto, suas expectativas em relação a novas

[1] Segundo WHEELRIGHT e CLARK (1992), é o tempo transcorrido desde o conceito inicial de um produto até a sua introdução no mercado.

características crescem. O mesmo fenômeno acontece com as expectativas de desempenho dos produtos. Exemplo disso são os mercados de computadores e automóveis. Para PATTERSON (1993), os produtos envelhecem rapidamente, o entusiasmo dos consumidores se esvai e a demanda cai. HARTLEY (1998) e EPPINGER (1998) sugerem que a complexidade dos produtos é crescente, em função da disponibilidade de tecnologia e também por exigência dos consumidores.

Ao mesmo tempo em que as fronteiras geográficas são eliminadas com o efeito da globalização das empresas, os setores de desenvolvimento afins dentro das várias subsidiárias das corporações globais se complementam. Em alguns casos, projetos são conduzidos simultaneamente por equipes dispersas geograficamente, mesmo que apresentem grandes diferenças culturais e lingüísticas (BAAKE, HAUßMANN e STRATIL, 1998).

Um dos fatores determinantes para a sobrevivência das empresas tem sido a sua capacidade de adaptação às novas exigências por parte dos clientes ou consumidores no sentido de lançarem produtos inovadores mais rapidamente. Uma vez que a obsolescência de determinados produtos tem sido alcançada de forma cada vez mais acelerada, as empresas têm sido pressionadas a lançar novos produtos e versões em intervalos de tempo cada vez mais breves, o que traz uma série de implicações no processo produtivo.

Segundo PATTERSON (1993), uma das características mais importantes das empresas do setor de produção de bens com alto teor tecnológico é o seu posicionamento perante a concorrência para o desenvolvimento de produtos inovadores. Tais empresas podem se posicionar como líderes em produtos inovadores, o que lhes confere grande vantagem, uma vez que a concorrência se posiciona como seguidora, trabalhando principalmente para a redução de preços, sem inserir maiores funcionalidades ou benefícios ao consumidor. Para as empresas líderes, a maior parte do lucro é decorrente das novas linhas de produtos, e não das antigas. Isto significa que o fluxo de lançamento de produtos deve ser contínuo, apoiado pela busca por inovação e antecipação das necessidades do cliente.

Para SMITH e REINERTSEN (1991), o ritmo de introdução de novos produtos depende basicamente das oportunidades de mercado que surgem. Sob uma perspectiva temporal, tais ocasiões são denominadas **janelas de oportunidade**. Levam vantagem as empresas que conseguem alinhar seus cronogramas de desenvolvimento e produção às janelas, mesmo que isto implique em custos de desenvolvimento maiores. As empresas que investem em desenvolvimento tecnológico percebem que as reduções de lucro na introdução de novos produtos são muito mais afetadas pelos atrasos no desenvolvimento do que devido a custos de produção que superam os valores planejados (WHEELWRIGHT e CLARK, 1992; HUNDAL, 1998).

Dentro deste cenário, a **Engenharia Simultânea** (ES) (WINNER, 1988) surgiu no final dos anos oitenta como uma proposta de encaminhamento do problema do desenvolvimento acelerado de produtos, integrando os diversos setores das empresas e reunindo alguns métodos que há décadas circulam no meio produtivo, como QFD, PDM, DFM/DFA, entre outros (PRASAD, 1996b; HARTLEY, 1998). Percebe-se que, ao longo do tempo, cada segmento interessado no assunto ES adota uma abordagem diferente dos demais. Desta forma, os fabricantes de *software* anunciam suas **ferramentas** como aquelas de ES, os acadêmicos procuram anunciar **métodos** sistemáticos para implementar ambientes de ES e as empresas que promovem o **comportamento** orientado à colaboração entre setores para o desenvolvimento de produtos anunciam que já vêm praticando ES.

1.2 FORMULAÇÃO DO PROBLEMA

Tudo leva a crer que realmente a ES seja a sistemática mais apropriada para o desenvolvimento rápido de produtos. Infelizmente, constata-se que não é ainda maciçamente utilizada (SMITH e REINERTSEN, 1991).

Segundo TANG, JONES e FORRESTER (1997), muitos fatores presentes nas corporações dificultam a completa implementação de um ambiente de ES. Sob o ponto de vista da estrutura organizacional e maturidade dos processos, há muitas

empresas despreparadas para aplicarem os preceitos desta sistemática. Além disso, os indivíduos envolvidos direta ou indiretamente no desenvolvimento de produtos, muitas vezes não estão suficientemente esclarecidos a respeito da importância da ES, ou até mesmo convencidos de sua eficácia (GOLDENSE, 1997a).

Muitas empresas que desenvolvem produtos de cunho tecnológico acusam dificuldade na prática da ES (LOVE, 2003). Para TANG, JONES e FORRESTER (1997), algumas destas empresas não estão sequer preparadas para a sua real implementação, visto que não estão estruturadas para tanto e, outras nem mesmo atingiram um nível suficiente de maturidade em seus processos de desenvolvimento de forma que se convençam da necessidade do emprego desta sistemática.

Uma vez suplantadas estas dificuldades, acredita-se que um sistema de informação que contemple todas as abordagens da ES possa servir como catalisador para a sua implementação, assim como possa produzir um efeito determinante para o estabelecimento de procedimentos internos nas empresas, cuja adoção provoque a disseminação dos conceitos, explícita ou implicitamente nas atividades rotineiras dos projetistas.

Por outro lado, há dúvidas a respeito do impacto oriundo da aplicação da tecnologia da informação sobre a sua implementação (KAPPEL e RUBENSTEIN, 1999). Acredita-se que as ferramentas existentes, se aplicadas isoladamente, não contemplam todos os aspectos relevantes desta sistemática de trabalho, de forma que, segundo EASON (1988), a imposição destes sistemas aos participantes do esforço pelo desenvolvimento de novos produtos pode ser mais prejudicial que benéfica. LIANG et al. (1994) supõem que tais ferramentas não reflitam o processo social envolvido no DIP, afastando o indivíduo de um esforço coletivo pela sua utilização, pois deveriam ser concebidos de forma a facilitar o trabalho entre membros geograficamente distribuídos, porém com objetivos comuns e especialidades complementares.

Para GATENBY (1994), antes de haver a ES, deve existir a Engenharia Colaborativa, isto é, a comunicação entre os indivíduos responsáveis pela execução de um projeto deve ser facilitada. Além de ser muito mais simples que a aplicação de

tecnologia da informação, nada substitui os contatos pessoais entre projetistas e demais colaboradores. Portanto é condição necessária para o sucesso da ES nas empresas que quaisquer barreiras entre as pessoas sejam eliminadas (PARKER, 1994).

Há na literatura várias iniciativas que procuraram tratar da aplicação de sistemas de informação que suportam equipe de desenvolvimento (MANSFIELD et al., 1999; KIM e CHUNG, 1998; BAILEY e RUCKER, 1998; MAHER e RUTHERFORD, 1997; CHEE e GOH, 1998; THRONICKE e SCHOLZ, 1998; ANUPAM e BAJAJ, 1994; DANGELMAIER et al., 1999). Entretanto carecem de uma visão mais abrangente no sentido de incorporar fatores tais como o uso de métricas e suporte à captura do conhecimento, igualmente importantes ao sucesso em ambientes de ES. Esta pesquisa almeja contribuir para a evolução conceitual destas ferramentas, ao propor a incorporação de conceitos de ES além daqueles mais difundidos, como compartilhamento de dados e paralelismo de atividades de projeto.

Uma vez adotados os fundamentos da ES para o desenvolvimento de novos produtos e reconhecido que as ferramentas da atualidade ainda não contemplam as diferentes abordagens da ES para o desenvolvimento de produtos, esta pesquisa questiona como as diferentes visões de integração, controle, gestão e projeto auxiliado por computador poderiam ser reunidas numa única estrutura que suporta o DIP.

1.3 OBJETIVOS

1.3.1 Objetivo geral

A presente pesquisa tem como objetivo geral desenvolver uma forma de estruturação para o trabalho de equipes de desenvolvimento de projetos de produto, baseada nos preceitos da ES.

1.3.2 Objetivos específicos

Nesta pesquisa existem os seguintes objetivos específicos, alinhados com o objetivo geral determinado:

a) definir um formato e uma sistemática para captura de modelos de informação[2] que contemplem diferentes níveis de abstração para o desenvolvimento de produtos, assim como a sua decomposição em subsistemas;
b) conceber uma plataforma de trabalho computacional para o Desenvolvimento Integrado de Produtos[3] (DIP) que considere os componentes de tecnologia de informação disponíveis na atualidade;
c) propor um sistema de métricas[4] que avalie o desempenho de equipes em atividades conduzidas conforme a sistemática da ES.

1.3.3 Pressupostos

Com base nas evidências da literatura pesquisada, são definidos os seguintes pressupostos para a realização desta pesquisa:

a) que a ES seja a sistemática mais indicada para a redução de custos e prazos para o desenvolvimento de produtos tecnológicos, à exceção dos produtos desenvolvidos sob encomenda;

---

[2] Trata-se da maneira como dados de projeto primários (desenhos, listas de componentes, esquemas e outros) e secundários (atas de reunião, mensagens de *e-mail* e outros) são organizados, integrados e sincronizados de forma a facilitar a gestão das informações durante o ciclo de desenvolvimento do produto.

[3] Consiste num conjunto de ferramentas computacionais com funções específicas, porém integradas no sentido de permitir a execução, acompanhamento e administração do projeto de produtos segundo os preceitos da ES.

[4] São critérios pré-estabelecidos de avaliação de desempenho que permitem a gestão de um projeto ao conduzirem o processo de tomada de decisão fundamentada em indicadores.

b) que as empresas que se proponham a adotar a ES como sistemática de trabalho estejam estruturalmente organizadas para tal e que tenham atingido níveis suficientes de maturidade referentes ao processo de desenvolvimento, refletidas em sua capacidade de repetibilidade e monitoramento de processos;

c) que os indivíduos envolvidos no DIP estejam suficientemente convencidos da importância da ES no contexto atual de competitividade das empresas e de sua eficácia.

d) que o sistema proposto:

- reflita o ambiente social em que os projetistas se inserem;
- favoreça o trabalho colaborativo sob todos os aspectos propagados pela ES;
- não impacte o processo habitual de desenvolvimento sob a forma de ônus adicional de trabalho aos colaboradores, nem como forma de cerceamento de sua criatividade;
- utilize tecnologias já disseminadas pelo fato de tornarem qualquer implementação financeiramente viável, até mesmo para pequenas empresas e;
- envolva também a cadeia de fornecedores, clientes e parceiros tecnológicos, mesmo que dispersos geograficamente.

Desta forma, apresenta-se a seguinte tese: é possível construir uma plataforma de suporte ao desenvolvimento acelerado de produtos, baseada em ferramentas computacionais, que facilite a implantação dos conceitos da ES, considerando-se todas as suas abordagens.

1.4 JUSTIFICATIVA

Uma vez aceito o paradigma apresentado pela ES como uma alternativa para os problemas da competição global e crescente demanda por produtos melhores, de

custos reduzidos e em prazos mais curtos, é natural que se indague a respeito das dificuldades de sua implementação nas empresas. Dentro deste contexto, esta investigação questiona sobre a estrutura de trabalho mais adequada para a efetiva implementação dos preceitos da ES.

Dentre as possíveis causas para o fenômeno da dificuldade de implantação dos conceitos da ES, do ponto de vista da aplicação da tecnologia da informação, destacam-se:

a) as ferramentas atuais para o trabalho colaborativo não são aceitas pelos projetistas pelo fato de não representarem o que socialmente acontece durante o desenvolvimento de um produto por uma equipe e;
b) há várias ferramentas que se sobrepõem em sua funcionalidade, portanto ainda não fornecem isoladamente uma plataforma de trabalho que contemple todas as abordagens da ES.

De forma análoga, a adoção de plataformas CAD nos anos oitenta e noventa levou o desenvolvimento de produtos de um patamar a outro quanto à sua eficiência e rapidez. A partir de sua utilização maciça, criaram-se outras expectativas de desempenho, realidade esta permitida também em função de uma quebra de paradigma. Atualmente espera-se muito mais de um ambiente de desenvolvimento baseado em ferramentas CAD do que quando o modelo pressupunha a utilização de desenhos elaborados manualmente, mapotecas e setores de arquivamento de desenhos. Portanto, acredita-se que a plataforma proposta nesta pesquisa permita a consolidação das práticas de ES, promova o ser humano para funções mais criativas e menos burocráticas e resulte em novas expectativas quanto aos prazos e custos de desenvolvimento de produtos.

A plataforma proposta deverá permitir que a integração dos membros das equipes de desenvolvimento, a utilização de métricas para controle e a introdução de ferramentas computacionais promovam uma mudança de paradigma no comportamento dos envolvidos no desenvolvimento de produtos. Desta forma, passariam a considerar outras questões igualmente relevantes para a aceleração dos

ciclos de desenvolvimento, tais como a capacidade de geração de idéias para produtos inovadores, o gerenciamento eficaz do *portfolio* de produtos das empresas e o melhoramento dos mecanismos que permitem a rápida e eficaz análise das novas idéias e seu desencadeamento em atividades de desenvolvimento, a fim de aproveitar melhor as oportunidades que mercado consumidor oferece.

2 REVISÃO DE LITERATURA

2.1 HISTÓRICO

O setor produtivo tem passado por constantes mudanças nas últimas décadas, sempre objetivando a redução do tempo entre a solicitação do cliente e a entrega efetiva do produto. Na década de setenta a área correspondente à entrega de produtos aos clientes passou por profundas alterações. A logística e as redes de distribuição foram aprimoradas de forma que a redução nos tempos de entrega passou a ser um quesito fundamental na escolha de um produto. Na década de oitenta houve a revolução da manufatura. A introdução de técnicas japonesas como *Just-in-Time* e *Kanban* entre outros, modificaram drasticamente os *layouts* de chão de fábrica (WOMACK, JONES e ROOS, 1992; ANDERSON, 2003). Os estoques intermediários foram reduzidos, pois consumiam recursos preciosos. As demandas nos revendedores passaram a programar as linhas de produção, de forma que não se produzisse o que não fosse vendido.

Segundo SMITH e REINERTSEN (1991), uma terceira onda de revoluções está acontecendo, a qual consiste na redefinição dos processos de desenvolvimento, a fim de acelerá-los. Alguns termos como **Engenharia Simultânea**, **Engenharia Paralela**, **Engenharia Colaborativa**, **Desenvolvimento Rápido de Produtos** e **Gerenciamento do Ciclo de Vida do Produto (PLM)** passaram a fazer parte do quotidiano de gerentes de projeto e projetistas desde meados dos anos oitenta. GOLDENSE (1997a), PRASAD (1996a), e KHURANA e ROSENTHAL (1997) acreditam que a próxima revolução enfocará a criação e definição dos produtos, e que a seguinte atingirá o nível das decisões gerenciais nas empresas, sempre com o objetivo final de reduzir o tempo de atendimento à demanda.

Neste cenário, em que o trinômio **tempo-qualidade-custo** domina o dia-a-dia das empresas, até por uma questão de sobrevivência, é preciso compreender o

desenvolvimento de produtos, seja de artefatos, seja de *software*, como um processo a ser depurado e otimizado, assim como aconteceu com os processos de administração da manufatura dos anos oitenta (ROY, USHER e PARSAEI, 1999).

EVBUOMWAN e ANUMBA (1996) citam que em dezembro de 1988 foi publicada a definição atualmente aceita para "ES", no IDA: "ES é uma abordagem sistemática para o projeto integrado e simultâneo de produtos e seus processos correlatos, incluindo manufatura e pós-venda". Este procedimento visa fazer com que as pessoas envolvidas no desenvolvimento considerem todos os elementos do ciclo de vida de um produto, desde a concepção até o descarte, incluindo qualidade, custo, prazos e necessidades do cliente.

Nas entrelinhas desta definição pode-se perceber que não é suficiente que apenas a tradicional Engenharia do Produto seja mobilizada para a implementação dos conceitos de ES. Representantes das várias áreas que agreguem conhecimento e experiência ao produto podem e devem ser chamados a participar (CASAROTTO FILHO; FÁVERO e CASTRO, 1999). Vem daí a necessidade da formação de equipes interdisciplinares coordenadas e voltadas a um objetivo final: satisfazer as necessidades do cliente, o que sem dúvida alguma trará retorno financeiro às empresas. *Marketing*, vendas, assistência técnica, testes, fabricação, engenharia, expedição e demais áreas de conhecimento devem ser envolvidas, se não em dedicação completa, pelo menos parcialmente. Em função da inclusão destas áreas nas definições dos produtos, e por nem todas serem exercidas por engenheiros de carreira, na segunda metade da década de noventa optou-se pelo termo **Desenvolvimento Integrado de Produtos (DIP)**, em contraste às restrições que o termo Engenharia Simultânea trazia implicitamente (GOLDENSE, 1997a).

O envolvimento da cadeia de fornecedores no sentido de partilhar riscos e responsabilidades é hoje uma realidade de extrema importância no cenário produtivo (KYRATSIS e MANSON-PATRIDGE, 1999). Na medida em que o mercado impõe reduções significativas nos custos dos produtos, cujos componentes fixos e variáveis têm sido revistos e minimizados de forma a responder a estas pressões, fornecedores

mais qualificados e com menores custos têm sido buscados, a ponto de hoje participarem como parceiros com metas de trabalho, a médio e longo prazos.

Por outro lado, ES e Engenharia de Sistemas são sistemáticas exaustivamente discutidas nos últimos anos, que visam o desenvolvimento de novos produtos. Claramente a Engenharia de Sistemas surgiu da necessidade de tratamento do projeto de produtos complexos, preocupando-se com os aspectos da subdivisão do trabalho em problemas mais fáceis de serem resolvidos. Apesar de haver uma forte tendência em se utilizar o termo ES para as questões referentes ao trabalho interdisciplinar em equipes, e o termo Engenharia de Sistemas para as questões referentes ao desenvolvimento de produtos de alta complexidade, acredita-se que não haja muita distância entre ambos. Em outras palavras, segundo GARDINER (1996), a ES pode se beneficiar da disciplina rígida da Engenharia de Sistemas, assim como a Engenharia de Sistemas pode ser favorecida em algumas circunstâncias pela quebra judiciosa de regras. Os proponentes da ES enfatizam o aspecto *time-to-market*, ao passo que aqueles da Engenharia de Sistemas enfatizam os aspectos da qualidade, particularmente a importância de se verificar se todos os requisitos do cliente ou consumidor foram efetivamente incorporados ao produto final.

Em busca de competitividade, os custos de mão-de-obra das empresas, tanto nas fases de produção quanto no desenvolvimento do produto, têm sido racionalizados. Isto traz conseqüências importantes nas empresas, especialmente em setores tradicionais como a Engenharia do Produto, na medida em que seu sucesso depende da alocação eficiente dos recursos humanos disponíveis e da adoção de ferramentas de projeto que favoreçam questões como reusabilidade e compartilhamento de informações. Em virtude desta constatação, muitas empresas têm evoluído no tocante à sua estrutura organizacional, deixando de ser estruturas fundamentalmente departamentalizadas para se transformarem em organizações projeto-orientadas.

Além da distribuição racional de recursos humanos, percebe-se que a gerência de desenvolvimento exerce papel decisivo na adoção de práticas que favoreçam a integração das várias áreas de conhecimento envolvidas no

desenvolvimento de um novo produto. Acredita-se que a própria formação dos engenheiros gerentes, segmentada em disciplinas, tenha colaborado para a consolidação de estruturas departamentalizadas nas empresas (SMITH, 1997b).

O estabelecimento de uma equipe de trabalho para um determinado projeto colide com a estrutura organizacional da maioria das grandes empresas. Ao longo do tempo as empresas se muniram de organogramas complexos que valorizam o conhecimento segmentado e não integrado. Foram construídas barreiras departamentais, que fixam as responsabilidades. Uma vez cumprida uma determinada etapa de um projeto num setor, os resultados eram transferidos para a etapa seguinte, sem responsabilidade ou comprometimento com o sucesso final do projeto.

Esta mentalidade trouxe enormes prejuízos para as empresas porque a falta de comunicação entre os setores gerava a necessidade de retrabalhos, mesmo em fases avançadas do desenvolvimento e fabricação de produtos. A mudança de cultura no meio industrial passa pelo amadurecimento dos participantes, ao se comprometerem com a realização dos projetos de forma integral. O reconhecimento profissional deveria ser baseado nestes resultados e não no conhecimento específico de um indivíduo.

Uma geração posterior de organogramas corporativos trouxe como grande inovação a coexistência entre os departamentos funcionais (onde os especialistas se reúnem) e as equipes interdisciplinares de projeto (MELLO, 1997). Nesta concepção cada um contribui com o que tem de melhor. Isto significa que a autonomia das equipes está garantida quanto à tomada de decisões e definição de recursos (humanos e materiais), mas os grupos funcionais ainda existem, mantendo uma espécie de lastro do conhecimento. Ao encerrar a sua participação numa equipe de projeto, o colaborador retorna ao grupo funcional ao qual primariamente pertence e realiza atividades de treinamento de pessoal, trabalhos independentes e melhorias de processo. Por exemplo, um projetista poderia trabalhar na adequação de suas ferramentas de trabalho para torná-las mais produtivas (CLAUSING, 1997; CASAROTTO FILHO; FÁVERO, 1999).

Dois fatores nesta nova concepção são preocupantes para as empresas que procuram implementá-la. Primeiramente, os gerentes dos grupos funcionais invariavelmente perdem poder. Então pessoas que tinham planos de alcançar estas posições em suas carreiras observam que seria melhor se firmarem como bons gerentes de projeto do que como gerentes funcionais. Os parâmetros que as empresas adotariam para recompensarem os colaboradores em função de sua produtividade tornam-se confusos ou, no mínimo, questionáveis. E, finalmente, a partir de uma situação em que cada colaborador está envolvido em vários projetos simultaneamente, disputando o tempo disponível, há dificuldade para as empresas se prepararem para a outra situação em que há dedicação integral para a realização de um projeto em específico (KERZNER, 1995).

O individualismo apresenta-se como uma das mais marcantes características do meio industrial, não só no Brasil como nos países ocidentais. Percebe-se que a carreira profissional esteja vinculada com o sucesso individual das pessoas, mesmo que isto não traga resultados no âmbito coletivo. Desta forma os objetivos das empresas ficam colocados em segundo plano. A isenção de comprometimento quanto ao sucesso dos projetos traz grandes prejuízos às empresas, a partir do momento em que os projetistas somente assumem responsabilidade sobre as partes do projeto que lhes cabem.

As culturas orientais, em especial a japonesa, apresentam grandes diferenças com relação às ocidentais no que tange à participação e realização do indivíduo, dentro do contexto da ES. Nestes países o indivíduo se sente recompensado com o sucesso de sua equipe, sem restrições. Desta maneira parece óbvio o motivo pelo qual os países ocidentais têm observado tanta resistência na introdução dos conceitos de ES. Isto tanto é verdade que o termo **Engenharia Simultânea** foi cunhado no ocidente, uma vez que no oriente trata-se de uma conseqüência natural dos seus valores e da maneira como trabalham.

Em virtude de fatores sócio-antropológicos, cultura e tradição, dificilmente os países ocidentais poderão alterar seus padrões comportamentais na sua plenitude.

Pode-se, entretanto, adaptar os princípios da ES à realidade de cada país. Neste sentido, deve-se fazer uma diferenciação entre Brasil e Europa, ou Brasil e Estados Unidos. As nuanças culturais de cada localidade devem estar presentes nas empresas, pois são indivíduos cuja formação é profundamente influenciada pelo meio cultural em que se vive.

Dentre as melhorias implementadas nas empresas nos últimos anos, o esforço pela adoção de medidas para o aprimoramento da qualidade de produtos e processos é evidente. São óbvios os benefícios desta iniciativa, porém há de se observar que o interesse das empresas pode estar, única e exclusivamente, no enquadramento de seus procedimentos a normas ISO ou similares, tendo em vista a abertura de possibilidades de acordos comerciais e parcerias, e não explicitamente para que o trabalho seja executado segundo padrões reconhecidos de qualidade.

Paralelamente aos aspectos organizacionais e humanos, a evolução das ferramentas computacionais disponíveis para o desenvolvimento de projetos de Engenharia tem modificado substancialmente a maneira segundo a qual os engenheiros trabalham. Depois de praticamente duas décadas de evolução, as ferramentas de projeto e análise auxiliadas por computador (CAD, CAE) estão hoje mais intuitivas. Similarmente, a experiência dos usuários com a aplicação de ferramentas computacionais tem avançado e os procedimentos e documentações de projeto têm sido adaptados ao trabalho com computadores. Tais aplicativos permitem atualmente um grau de reaproveitamento, recuperação e compartilhamento de informações anteriormente nunca vistos na área de desenvolvimento de produtos. Entretanto, projetistas são pessoas e portanto nem sempre o potencial das ferramentas é aproveitado na sua totalidade. Segundo KAPPEL e RUBENSTEIN (1999) o trabalho criativo não tem sido tão beneficiado pela introdução destas ferramentas como poderia se supor. Além disso pressupõe-se que os engenheiros desejem a integração com outras áreas, não só que as ferramentas possibilitem tecnicamente este trabalho.

A partir dos anos noventa, com a popularização dos microcomputadores e a interligação de redes de computadores, a aplicação de sistemas de informação em

ambientes corporativos foi consolidada. Mesmo em pequenas empresas já há condições, e principalmente demanda, para que os sistemas de informação sejam implementados. Tal medida proporciona um meio seguro e rápido para o acesso às informações que circulam nas empresas. A redução do fluxo de documentos em papel favorece a atualização das informações em tempo real e a rapidez de distribuição.

É importante ressaltar que há diversas modalidades de ferramentas que poderiam compor sistemas de informação que suportam o trabalho em equipe. Cada qual tem funções específicas, vantagens e desvantagens. Algumas dão ênfase à coordenação de atividades, permitindo a modelagem de um determinado processo, caracterizando-se pelo rigor com que impõem restrições às informações e procuram controlar o fluxo de dados entre os membros da equipe. Há ferramentas para o gerenciamento de projetos, que estão orientadas ao planejamento e controle das atividades, facilitando o gerenciamento de recursos humanos, temporais e materiais. Há também ferramentas cuja finalidade é organizar os dados de projeto, permitindo que informações primárias (desenhos, diagramas e outros) e secundárias (informações que constam em atas, mensagens de correio eletrônico, propostas comercias de fornecedores, entre outros) sejam disponibilizadas e até mesmo sincronizadas. Finalmente há os sistemas de CSCW, que passaram a despertar o interesse entre pesquisadores e usuários a partir dos anos noventa, quando foi detectada a necessidade de ferramentas que objetivassem a definição de uma plataforma de acesso a dados e trabalho colaborativo. Na verdade, por se fazer necessário um certo rigor de linguagem, tais plataformas objetivam sim o trabalho cooperativo, uma vez que se presume que o termo colaboração não transmita o caráter de simultaneidade envolvido no desenvolvimento de um produto.

2.2 FUNDAMENTOS

Apesar da terminologia ser recente, sabe-se que os fundamentos da ES podem ser atribuídos a várias iniciativas, que datam desde o início do século XX. O

primeiro fundamento, segundo SMITH (1997b), é a participação crescente das definições dos processos de fabricação em decisões acerca do projeto de produto. A consideração destes fatores durante as primeiras etapas de desenvolvimento cria oportunidades para a redução de custos de desenvolvimento e melhoria da qualidade do produto; a sua omissão, por outro lado, geralmente é causada no ambiente organizacional por barreiras funcionais entre projeto e fabricação.

Um segundo fundamento da ES se refere à formação de equipes interdisciplinares para o desenvolvimento de produtos. Objetiva realizar a integração entre diversas áreas de conhecimento nas empresas, tais como *marketing*, vendas, assistência técnica, entre outras. Para SMITH, a idéia de integrar todas estas áreas também não é recente.

A ênfase às necessidades dos clientes durante o processo de desenvolvimento, também é considerada um fundamento da ES. Assim como a eliminação das barreiras corporativas interdepartamentais, é importante que a barreira entre clientes e projetistas seja transposta para que o verdadeiro ambiente e espírito da ES sejam atingidos (INTERNATIONAL SOCIETY FOR PRODUCTIVITY ENHANCEMENT, 2003; SOCIETY OF CONCURRENT PRODUCT DEVELOPMENT, 2003).

Não menos importante que os três fundamentos anteriores, a utilização do *lead time*[5] como vantagem competitiva, não é recente na literatura. Ao reduzi-lo, uma empresa é capaz de responder mais rapidamente às tendências do mercado ou incorporar novas tecnologias.

Dentre os preceitos da ES, o paralelismo parcial ou completo entre atividades ocupa uma posição de destaque (HAUPTMAN, 1996). Em função da departamentalização das grandes empresas, oriunda da concentração das

[5] *Development Lead Time*: é o tempo transcorrido desde a concepção de uma idéia até a aprovação do produto para produção; *Lead Time*: é o tempo total necessário para completar uma unidade de um produto ou serviço (PRASAD, 1996a, p. 47).

especialidades em **silos** (CLAUSING , 1997), consolidou-se a prática da execução seqüencial de tarefas de um projeto, notadamente definição dos requisitos (*marketing*), desenvolvimento do produto (engenharia), definição do processo e fabricação (produção), distribuição (logística) e pós-venda (assistência técnica). Uma vez que a isenção de responsabilidade pelo sucesso do produto era constatada em todas as etapas do processo, era comum haver ciclos de reprojeto, causados por ajustes que se faziam necessários, em função da falta de envolvimento das especialidades nas etapas de definição do produto. A Engenharia do Produto não poderia mais ser exclusiva de engenheiros, já que outros profissionais teriam muito a colaborar. Além disso, em função das pressões sobre custo e *time-to-market*, era necessário abreviar a duração dos ciclos de desenvolvimento.

O paralelismo entre atividades só é viável através da integração entre áreas de conhecimento mediante a constituição de equipes interdisciplinares. Entretanto, para que se concretize, assume-se que as informações serão trocadas ao longo do desenrolar das tarefas concomitantes e, portanto, tratam-se de informações incompletas. A partir do momento em que estas informações, mesmo sendo parciais, são compartilhadas com outros membros da equipe, existe a possibilidade de algum retrabalho ser necessário em virtude de modificações implementadas. Este é, na verdade, um ônus que a empresa paga pela adoção da prática de ES, pois há o risco de se ter de repetir uma tarefa parcial ou integralmente em função das modificações. Quanto maior for a sobreposição entre tarefas, maior será o risco envolvido.

Segundo SMITH (1997b), apesar de indícios comprovados de que os fundamentos da ES não são recentes, o movimento mundial pela adoção destas práticas na década de oitenta parecia ter um caráter inovador. O autor sugere que a sua adoção somente nas últimas décadas do século XX pode ter ocorrido primariamente em função da redução das barreiras para a cooperação interdisciplinar através da tecnologia da informação. Alega ainda que a formação deficiente dos responsáveis pelo desenvolvimento de produtos em relação às melhores práticas existentes também possa ser atribuída como a causa deste fenômeno. Também acredita-se que a evidência

do sucesso das empresas japonesas em aspectos de qualidade e tempo de resposta ao mercado tenha pressionado as empresas norte-americanas e européias nos anos oitenta a revisarem seus processos e investigarem as causas da redução de sua competitividade no mercado (WOMACK , 1992).

A ES é, de fato, uma maneira de se chegar ao desenvolvimento acelerado de produtos. Entretanto, bastaria haver estimativas melhores e a introdução de métricas de controle no gerenciamento para que as metas de prazo fossem atingidas sem maiores atrasos. De acordo com AITSAHLIA, JOHNSON e WILL (1995), a prática do paralelismo entre atividades de projeto, que é um dos preceitos da ES, pode chegar a resultados piores que aqueles obtidos pelo seqüenciamento de atividades, em função do risco associado à troca de informações incompletas entre projetistas. Para GOLDENSE (1997b) a falta de estimativas precisas para o cumprimento de etapas do desenvolvimento é que culmina no descumprimento dos cronogramas estabelecidos. Portanto, é necessário que as empresas entendam o desenvolvimento de um produto como um processo, cujo monitoramento deve ser efetuado através de métricas, da mesma forma como processos de chão de fábrica são medidos em grandezas físicas, tais como o comprimento e o peso, ao longo de sua execução.

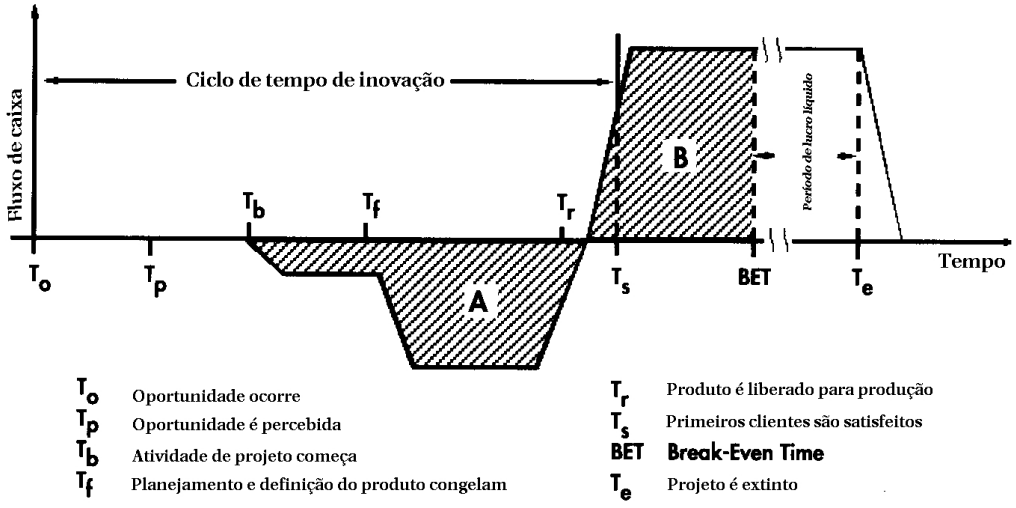
A falta de previsibilidade no processo de desenvolvimento tem suas raízes nos perfis dos projetistas e engenheiros do produto. Como se trata de um trabalho criativo, muitas vezes encarado até mesmo como uma extensão da personalidade dos envolvidos, qualquer medida que cerceie seus trabalhos é recebida com restrições e resistência (KAPPEL e RUBENSTEIN, 1999). Desta forma, esta tarefa passa a exigir uma mobilização em nível gerencial no sentido de esclarecer os benefícios da previsibilidade e repetibilidade de processos, para o próprio bem dos engenheiros de desenvolvimento.

2.3 CICLO DE VIDA DE PRODUTOS

Segundo PATTERSON (1993), produtos são definidos como bens, ou seja,

objetos tangíveis que se pode ver, tocar e usar. O processo de desenvolvimento de novos produtos depende tanto do fluxo de informações quanto do fluxo de materiais (CLARK e FUJIMOTO, 1995). Antes de um consumidor desempacotar um bem, como um computador, ou instalar uma máquina embaladora de alta velocidade, e muito antes de um novo carro sair da linha de montagem, o produto (ou alguma versão preliminar dele) começa com uma idéia. Em seguida, esta idéia ganha corpo e progressivamente formas mais detalhadas e concretas: idéias em esboços, esboços em desenhos, desenhos em protótipos, e assim por diante até que um produto acabado sai da fábrica. Quando está finalmente nas mãos do consumidor, o produto é convertido novamente em informação, pois é a realimentação do consumidor que pode provocar alguma modificação no mesmo.

Figura 1 - Ciclo de vida para a introdução de novos produtos.



Fonte: Adaptado de PATTERSON (1993).

Há dois objetivos básicos em qualquer negócio que pretenda alcançar lucro: satisfazer os clientes e obter retorno do investimento. O fator fundamental para se alcançar estes objetivos é o tempo consumido para se desenvolver e introduzir um novo produto, medido desde o momento em que a oportunidade para seu surgimento ocorre. Para visualizar a natureza deste impacto, considere-se a Figura 1,

correspondente ao ciclo de vida para a introdução de novos produtos (PATTERSON, 1993).

O tempo T_o é o momento em que a oportunidade para um novo produto ocorre. É um momento crucial no DIP, porém não é usualmente perceptível. Trata-se do instante em que uma tecnologia emergente se sobrepõe a uma necessidade do cliente e dispara a possibilidade de um novo produto. O tempo do ciclo de inovação de produtos é o período entre o momento em que a janela de oportunidade abre e o momento em que os primeiros clientes são satisfeitos. A oportunidade aparece e geralmente é seguida de alguma demora até o instante T_p, quando é percebida. É tarefa da empresa reduzir este atraso ao mínimo e obter um produto para aquela janela o mais rápido possível.

A Figura 1 traz de forma implícita que a oportunidade sempre precede o desenvolvimento, mas nem sempre isto é verdadeiro. Ocasionalmente o desenvolvimento de produtos pode, na verdade, iniciar antes da oportunidade. Neste caso, se os engenheiros têm uma idéia que antecipa a ocorrência de uma oportunidade de mercado, podem começar o desenvolvimento antecipadamente. Com alguma sorte, no momento em que a oportunidade ocorrer, pode-se lançar o produto que a explora. Por definição, no momento em que a oportunidade ocorre, as pessoas desejariam comprar o produto, se este estivesse disponível. Um atraso de um mês na introdução de um produto significa um mês de perda de receita. Além disso, se um produto chega ao mercado depois que um ou mais competidores tenham lançado seus produtos, não só terá havido perda de parte da janela de oportunidade do mercado, como também haverá perda de participação no mercado. Desta forma, o sucesso do negócio é uma função monótona decrescente do tempo de introdução do produto. Em outras palavras, quanto mais tarde uma empresa lançar um produto, menores serão as chances de sucesso.

A janela de oportunidade não é conhecida *a priori*. O instante da oportunidade pode ou não ser perceptível, mas o momento da sua extinção (T_e) é sempre obscuro. Este depende fortemente de fatores imponderáveis como a evolução

da tecnologia no tempo, a resposta dos competidores e os passos seguintes da própria empresa. Os únicos controles que a empresa tem sobre o momento da extinção são as escolhas que faz nas primeiras etapas do projeto, tais como mercado visado, características, tecnologia operacional, tecnologia de manufatura, faixa de preço objetivada, canais de distribuição, sistema de suporte ao cliente, e assim por diante.

Quando o planejamento congela, também o momento da extinção é congelado. Assumindo que esta seja a melhor implementação com os processos e tecnologia disponíveis naquele instante, não há nada que possa ser deixado para as demais empresas competidoras. Se quiserem competir, terão de criar algo substancialmente melhor com novas tecnologias ou processos.

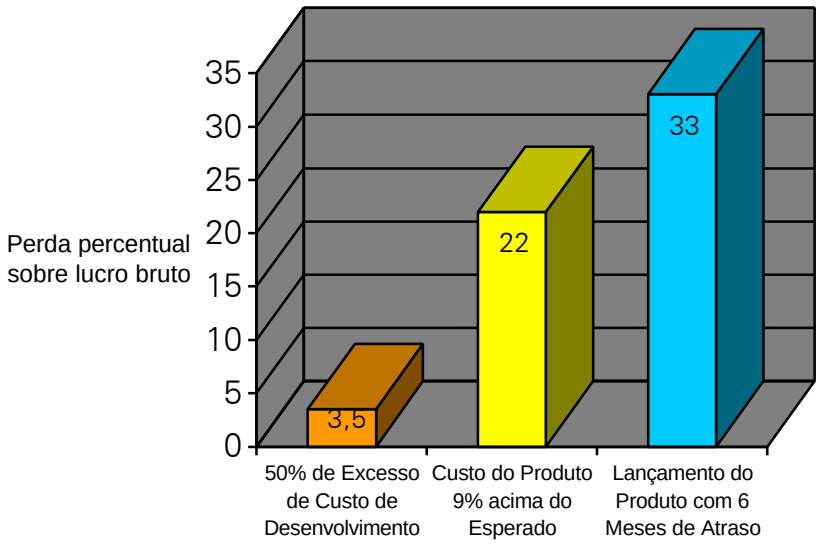
Dentro deste cenário, percebe-se que o ponto BET (*break-even time*), ilustrado na Figura 1, ocorrerá no momento em que as áreas A, correspondente ao investimento em desenvolvimento, e B, correspondente à receita advinda de vendas, se igualam. A partir daí, a área compreendida entre o nível de receita obtido e a abscissa T_e representará o período de lucro líquido a ser obtido.

Uma vez compreendido este fluxo de caixa, pode-se imaginar quais medidas poderiam ter algum resultado a fim de aumentar as possibilidades de faturamento corporativo. Uma alternativa seria a redução dos custos de desenvolvimento, o que sem dúvida reduziria a área abaixo do eixo das abscissas. Isto, em termos práticos, significa racionalizar as despesas de desenvolvimento. Uma vez racionalizados os custos, qualquer redução a mais pode provocar perda de produtividade e, portanto, passa a não ser interessante.

Outra maneira de se reduzir esta área seria antecipar o lançamento dos produtos. É aí que a ES tenta se posicionar como alternativa mais interessante, ao propor o paralelismo de tarefas, empregando simultaneamente o conhecimento de todas as áreas envolvidas com desenvolvimento em direção de um mesmo objetivo, que é o lançamento de um produto no mercado potencial, com o menor número possível de falhas. Obviamente, de acordo com esta estratégia, os gastos deverão aumentar, mas mesmo assim o processo é compensatório na medida em que os prazos

sejam reduzidos e a qualidade do produto seja garantida. Ao comparar o impacto de anomalias ocorridas ao longo do ciclo de desenvolvimento de produtos, alguns estudos indicam que, apesar de haver uma certa predisposição em se acreditar no que seja mais facilmente mensurável, o atraso no desenvolvimento de produtos pode trazer conseqüências muito mais graves do que custos elevados dos produtos em relação ao planejado, assim como custos excessivos de desenvolvimento. A Figura 2, citada por GOLDENSE (1997a) e PATTERSON (1998), traz o efeito destas anomalias, sugerindo que, do ponto de vista das empresas, seja muito mais vantajoso lançar um produto mais caro no mercado do que lançá-lo tardiamente.

Figura 2 - Impacto de anomalias de desenvolvimento em termos de perda do faturamento corporativo (%).



Fonte: Adaptado de GOLDENSE (1997a).

2.4 FASES DO DESENVOLVIMENTO

De acordo com KERZNER (1995), o processo de desenvolvimento de um produto pode ser caracterizado por distintas fases, quais sejam:

a) Surgimento da idéia: consiste na investigação das necessidades, potencial e tecnologia no mercado, elaboração de um programa de

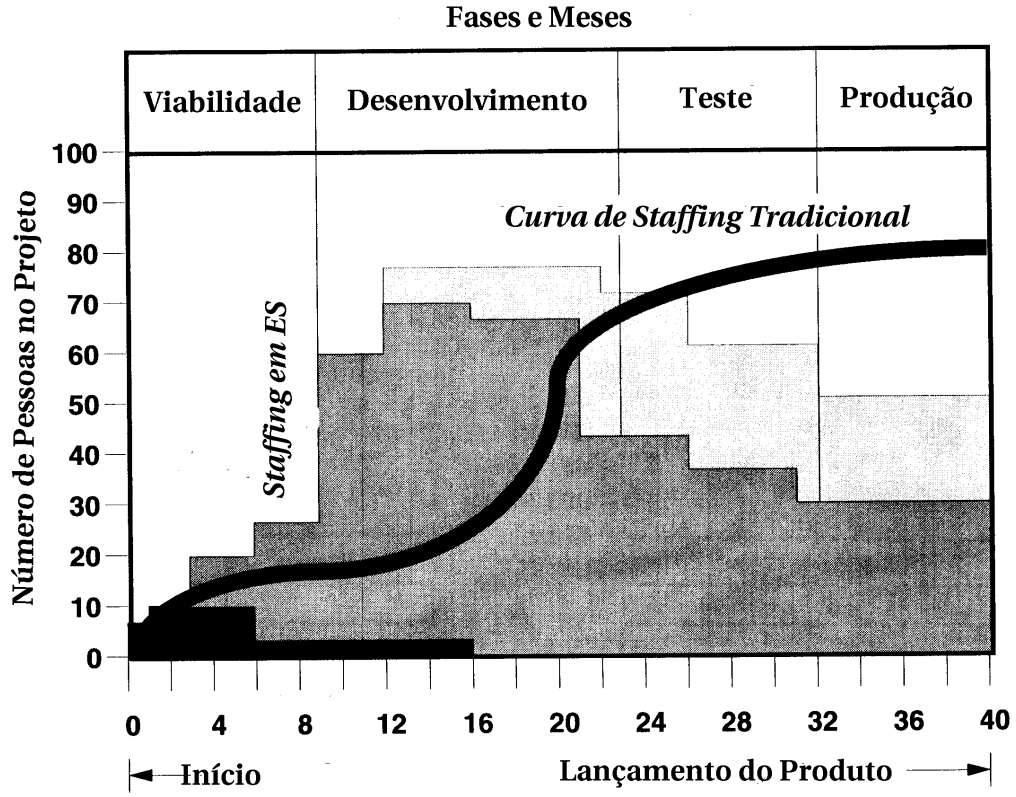
viabilidade e planejamento da definição do produto;

b) Definição do produto: consiste na definição e aprovação dos requisitos do cliente, definição e aprovação da solução proposta e planejamento do projeto para fornecimento da solução completa;
c) Implementação: consiste no projeto, teste e produção da solução completa. Por sua vez, esta fase pode ser subdividida em:
    - Concepção: consiste na conversão dos requisitos do cliente em especificações técnicas, de forma que se tenha como resultado o embrião do modelo de informações que servirá de base para a agregação de conhecimento a partir das várias funções envolvidas, desde *design* e detalhamento técnico até informações sobre fornecedores, reciclagem, serviços, entre outros;
    - Projeto: consiste especificamente na solução técnica que satisfará o cliente e seu detalhamento;
    - Prototipagem: consiste na materialização do detalhamento técnico, previamente à etapa de produção, de forma a facilitar a visibilidade das questões de funcionalidade, fabricabilidade, testabilidade, operabilidade e manutenibilidade;
d) Teste de aceite: consiste no teste e certificação de que a solução funciona no ambiente do cliente, lançamento do produto e introdução do produto juntamente com treinamento de pessoal de campo;
e) Produção: consiste no suporte ao programa de garantia, determinação de perdas e ganhos durante o processo de desenvolvimento e identificação das melhorias no processo.

Cada fase pode ser separada das demais através de algum mecanismo de aprovação, mediante reuniões ou assinaturas de responsáveis. Esta sistemática, denominada *Phase-Gate*, *Phase-Milestones* ou *Phase-Checkpoints* está disseminada e é aceita na indústria como uma forma adequada para se conduzir o projeto de um

produto (GOLDENSE, 1997a). Este mecanismo funciona como uma espécie de acordo ou compromisso com a equipe envolvida nas etapas subseqüentes de forma que as metas e a metodologia adotada são estabelecidas claramente.

Figura 3 - Curva de *staffing*[6].



Fonte: Adaptado de GOLDENSE (1997a).

Segundo GOLDENSE (1997a), apesar de transcorrido pouco tempo até as fases de concepção, projeto e prototipagem, é neste período que 90% do custo final do produto é definido. Desta forma, acredita-se que a aplicação de mais recursos humanos, tecnológicos e financeiros nestas etapas possa ser de grande valia no sentido de se reduzir o potencial de problemas relacionados com fabricabilidade, cadeia de fornecedores e assim por diante. Vem deste fator mais uma justificativa irrefutável

[6] *Staffing*: alocação de recursos humanos em projetos.

para a aplicação da ES.

Para GOLDENSE, até os anos noventa, a indústria ocidental diferenciou-se da oriental na medida em que tradicionalmente as equipes de desenvolvimento atuavam isoladamente daquelas responsáveis pela fabricação e comercialização dos produtos. A integração destas áreas acontecia de maneira tardia, após a etapa de prototipagem, quando eram evidenciados os problemas que mais tarde afetariam as demais áreas funcionais. Neste instante, muitas outras pessoas eram envolvidas no desenvolvimento, a fim de resolver os problemas gerados pela falta de comunicação e troca de informações. Entretanto, percebeu-se que os custos relacionados a estas operações de *redesign* evoluíam exponencialmente, conforme a fase em que eram acionados. É muito menos custoso resolver um problema no produto quando este está sendo projetado do que quando está sendo prototipado, que por sua vez é muito menos custoso do que resolvê-lo na fase de fabricação, e assim por diante. Acredita-se que a proporção entre custos de *redesign* nas etapas de projeto e fabricação seja de um para dez mil.

A indústria oriental, por sua vez, percebeu estas questões muito antes e promoveu naturalmente a integração das diferentes áreas de conhecimento, o que lhe rendeu a posição de liderança em muitos segmentos da indústria por vários anos. A Figura 3, citada por GOLDENSE, demonstra comparativamente o perfil de envolvimento de pessoal nas duas situações em função das etapas de desenvolvimento. Este mesmo perfil pode ilustrar a evolução do número de alterações de projeto ao longo das etapas de desenvolvimento.

2.5 INTEGRAÇÃO

O processo de desenvolvimento simultâneo é realizado por uma equipe interdisciplinar que integra especialidades (funções). Há cinco tipos de configuração organizacional para o desenvolvimento de produtos: estrutura funcional, gerente de projeto **peso-pena**, gerente de projeto **peso-pesado**, equipe de execução de projetos e

equipe independente de desenvolvimento de produtos (WHEELWRIGHT, 1992; CLAUSING , 1997; CASAROTTO FILHO; FÁVERO e CASTRO, 1999). Os primeiros dois tipos organizacionais enfatizam a especialização funcional e têm menos interesse no mundo competitivo e complexo de hoje. As últimas três destas configurações são todas referentes a equipes interdisciplinares, apesar de se diferenciarem substancialmente umas das outras.

A forma de organização por gerente de projeto **peso-pesado** ainda tem os membros em departamentos funcionais. A forma da equipe interdisciplinar é definida pelo gerente, que tem poderes suficientes de tomada de decisão, de modo que consegue que os membros se concentrem no produto e seus clientes.

Na equipe de execução de projetos, muitos dos membros pertencem a departamentos funcionais, mas são emprestados à equipe de desenvolvimento pelo período em que o projeto se desenvolve, ou pelo menos por uma fração significativa do mesmo. Isto proporciona ainda mais ênfase ao produto e seus clientes que nas estruturas de gerente de projeto **peso-pesado**. Este arranjo possibilita também que os projetistas e demais envolvidos retornem periodicamente a seus respectivos departamentos funcionais de origem a fim de manter sua competência funcional.

Nas equipes de desenvolvimento independentes, os membros não pertencem a departamentos funcionais. Seu papel dentro da empresa é participar de equipes de desenvolvimento. Isto gera um enfoque ainda maior no produto e seus clientes que na equipe de execução de projetos, apesar de tender a causar alguns problemas de planejamento de carreira aos participantes. Além disso, as equipes de desenvolvimento independentes apresentam três perigos em potencial: obsolescência funcional, baixo aprendizado entre as equipes e baixo desenvolvimento tecnológico. Uma maneira de se evitar estes perigos é dispor de uma organização para o desenvolvimento tecnológico, responsável pelo tratamento destas questões.

Se os produtos são bastante simples, o mercado muda apenas vagarosamente e a especialidade técnica é muito importante, então a organização funcional tem bons resultados. Entretanto, para produtos complexos, num mercado que muda rapidamente

com especialidades técnicas normais, a organização funcional tem se mostrado insatisfatória. É muito lenta e fraca ao tratar da tal complexidade. A tendência, desde a década de oitenta, tem sido no sentido de se adotar configurações de equipes de execução de projetos, ou equipes independentes.

Para produtos relativamente simples, as equipes interdisciplinares são suficientes. Entretanto, para produtos mais complexos, a estrutura de desenvolvimento deverá contar com uma equipe para cada subsistema. Para produtos ainda mais complexos é possível que a estrutura tenha características adicionais como, por exemplo, outras camadas e módulos que serão posicionados entre a chefia da engenharia e as equipes de subsistemas.

Uma extensão às equipes de subsistemas pode ser obtida desde que os líderes das mesmas também formem uma equipe para tomada de decisões. Por exemplo, recursos podem ser redirecionados de um subsistema para outro se necessário. A prática deste tipo de trabalho, em equipe interdisciplinar de alto nível, deixa ainda a desejar em função do sucesso de equipes interdisciplinares de subsistemas.

Recentemente tem se tornado importante estender o conceito de equipes interdisciplinares de forma que abranjam toda a cadeia produtiva. A tendência é que sejam realizadas parcerias entre os integrantes da cadeia, o que de certa forma transcende o conceito tradicional de empresa, para que contemple participantes de várias organizações (KYRATSIS e MANSON-PATRIDGE , 1999).

2.6 ABORDAGENS

Diversas abordagens surgiram ao longo do tempo, baseadas nas premissas da ES. MEYER (1990) apresentou de maneira pragmática a relação entre redução de *lead time*, redução de custos e melhoria de qualidade ao afirmar que a ES é uma sistemática de trabalho que procura otimizar o projeto de produtos e processos de fabricação para alcançar *lead times* reduzidos, além de custo e qualidade melhorados através da integração de atividades de projeto e fabricação e da maximização do paralelismo

entre práticas de trabalho.

Por outro lado, KANNAPAN e MARSHEK (1992) procuraram enfatizar a questão da integração entre áreas de conhecimento. A ES, no caso ideal, seria uma sistemática que traz à tona todas as preocupações através do ciclo de vida do produto de forma simultânea, ao longo de seu projeto. A estratégia de simultaneidade fornece uma oportunidade para tratar antecipadamente da fonte de conflitos entre agentes do desenvolvimento que representam os diferentes pontos de vista de disciplinas da engenharia, tais como funcionalidade, viabilidade mercadológica, fabricabilidade, manutenibilidade, entre outros.

Diferentemente de outros autores, CLEETUS (1992) propôs de maneira mais precisa a estratégia de troca de informações e sua periodicidade. Para ele, ES é uma sistemática que enfatiza a resposta às expectativas do cliente e incorpora valores coletivos de cooperação, confiança e compartilhamento, de tal forma que a tomada de decisões é conduzida através de grandes intervalos de trabalho em paralelo, por todas as perspectivas do ciclo de vida do produto, antecipadamente no processo, e de forma sincronizada por períodos relativamente curtos de troca de informações que conduzam a um consenso.

Na mesma linha de raciocínio, PAASHUIS e BOER (1997) apresentaram uma visão que não envolve necessariamente a utilização de tecnologia da informação, mas enfatiza o aspecto colaborativo e de integração. De acordo com suas observações, ES é uma prática que abrange uma grande gama de mecanismos de integração estratégicos, de processos tecnológicos e organizacionais, objetivando a estreita colaboração, comunicação antecipada e mais freqüente entre as funções envolvidas no projeto, fabricação e comercialização de novos produtos, e um certo grau de sobreposição das fases que constituem este processo.

Segundo EVBUOMWAN e ANUMBA (1996), a ES pressupõe as seguintes ações:

- análise apropriada e estabelecimento de requisitos do cliente e especificações;

- desenvolvimento de soluções conceituais que sejam modulares, fáceis de fabricar e montar;
- integração de processo de fabricação e projeto do produto que melhor combinem necessidades e requisitos;
- desenvolvimento de interfaces entre subsistemas do produto para levar em conta tolerâncias e projeto para a robustez;
- adoção de uma abordagem sistêmica para o desenvolvimento do produto e consideração pelo ciclo de vida integral do mesmo;
- enfoque continuado no melhoramento do produto e processo de fabricação;
- co-locação de equipes multifuncionais, quando possível, para facilitar a comunicação;
- redução do *lead time* e custo do produto e;
- paralelismo do processo de desenvolvimento.

De maneira concisa, HUTHWAITE (1994) propôs que a ES seja praticada através dos seguintes fundamentos:

- projeto é a diretriz primária de qualidade, custo e tempo;
- reconhecimento da necessidade de ampliar a força do projeto mais cedo, mais amplamente e mais profundamente e;
- equipes interdisciplinares são a chave para vislumbrar a equação do projeto total.

Na interpretação de TIBBITS e KEANE (1995), a ES se baseia nos seguintes princípios:

- orientação ao processo;
- abordagem por equipes;
- *empowerment*[7];

[7] Segundo PARKER (1994, p. 47), significa estar investido de autoridade para tomar decisões e implementá-las.

- comunicação aberta e;
- satisfação do cliente.

Segundo DOWLATSHAHI (1994), há cinco diferentes abordagens para a ES. São elas: sistemas de informação, *software design*, inteligência artificial; integração entre CAD e CAM; Engenharia para o ciclo de vida ou Engenharia para o ciclo de vida unificado; DFM e DFA; e mudanças organizacionais e culturais.

KYRATSIS e MANSON-PATRIDGE (1999) sugerem que haja dois caminhos preferidos: a abordagem baseada em trabalho em equipe e a abordagem baseada em computadores. Há implementações que priorizam este ou aquele aspecto.

Sob o ponto de vista da academia, SUSMAN e DEAN (1992) sugerem quatro abordagens para a ES: mecanismos de integração, processos de grupo, codificação/informatização e condições de tarefas. Dentro da categoria **mecanismos de integração**, os trabalhos podem ser enquadrados nos seguintes assuntos: fronteiras de equipes (fornecedores, clientes), redução de número de projetos assumidos, co-locação, *empowerment*, ajuste de metas e revisões formais, avaliação de desempenho e gratificação com base em equipes. A categoria **processos de grupo** traz os seguintes assuntos: facilidade de tomada de decisão, número de agentes influenciadores, comunicação de duas vias, sobreposição de tarefas (preemptivas, iterativas, distributivas), liberação e uso de informações incompletas e simultaneidade de atividades. Os trabalhos da categoria **codificação/informatização** englobam os seguintes assuntos: linguagem comum, heurística em projetos, co-locação virtual, tecnologia de interface, escopo de regras de projeto, velocidade de acesso a dados, programação por metas e integração de tecnologias computacionais. Por último, a categoria **condições de tarefas** traz os seguintes assuntos: inovação, risco, incerteza, complexidade, interdependência, projeto evolutivo, sensitividade de projetos e ciclo de vida antecipado/tardio de produtos.

De uma maneira geral, a literatura pesquisada indica que haja basicamente três preocupações para que a adoção da ES tenha os efeitos desejados sobre as questões de rapidez, qualidade e custo, quais sejam: integração antecipada e

coordenada entre as áreas de conhecimento relevantes ao desenvolvimento de produtos, organização da equipe sob o prisma de interdisciplinaridade e dedicação integral ao projeto e, enfoque sobre as necessidades do cliente. É interessante notar que muitas vezes não existe menção às ferramentas tecnológicas que de alguma forma acelerem ou facilitem este processo, o que leva a crer que são importantes, mas não essenciais para a implementação da ES no desenvolvimento de produtos.

2.6.1 Integração

A primeira abordagem trata dos aspectos relativos à integração entre áreas de conhecimento nas empresas objetivando o desenvolvimento de produtos. Tal integração é obtida ao formar-se uma equipe cujos membros têm dedicação integral ou parcial ao projeto de um produto e a coordenação é atribuída a um elemento que recebe poderes e autonomia para decidir a respeito de todas as questões relevantes para que o projeto transcorra normalmente (COOPER e KLEINSCHMIDT, 1993) (MELLO, 1997).

A idéia de cooperação interdisciplinar já existe há muito tempo, conforme aponta SMITH (1997b) em seus estudos. De acordo com ele, há várias justificativas potenciais pelas quais a aplicação de ES possa ser de importância: competição econômica, novos processos de produção, e um lead time reduzido. Nenhum destes fenômenos é novo. São reais, mas há muito tempo existem como justificativas para cooperação interdisciplinar.

O princípio de utilizar equipes se estende para o passado, na década de quarenta. Este princípio tem aparecido regularmente na literatura depois de seu surgimento original. Observa-se que vários autores das décadas de cinqüenta e sessenta integravam todo um repertório de práticas de ES. Estes trabalhos descrevem a importância dos processos de fabricação no projeto de produtos, reconhecem a necessidade de formalmente incorporar análise mercadológica no processo de desenvolvimento do produto, e sugerem abordagens organizacionais para a realização

da integração, tais como a criação de equipes interdisciplinares.

Os problemas da interdisciplinaridade são explorados por PAASHUIS, e BOER (1997) ao afirmarem em seus estudos que numa determinada empresa a maior contribuição para redução do *lead time* do desenvolvimento seria a melhoria da comunicação entre vendas, *marketing*, desenvolvimento, engenharia, logística e fabricação. As informações são geralmente incompletas, irrelevantes, incompreensíveis e, ocasionalmente, mesmo incorretas. É um fenômeno bastante conhecido que diferentes áreas funcionais tenham diferentes perspectivas, usem diferentes estruturas de referência e linguagens, e tal empresa não é exceção. Como conseqüência, diferentes áreas freqüentemente interpretam mal as informações, o que leva ao entendimento imperfeito e perda de tempo e energia. Ainda segundo estes estudos, naturalmente há algum risco associado à sobreposição de atividades independentes, mas isto pode ser reduzido através da padronização de métodos de trabalho, especificações e materiais, estratégias e objetivos claros, não ambíguos e bem comunicados, e estreita colaboração e comunicação freqüente.

A mera adoção do exercício da comunicação entre setores traz resultados significativos, mesmo que não seja realizado nenhum trabalho de re-escaloamento de tarefas. De acordo com este posicionamento de DWIVEDI, KULPA e SOBOLEWSKI (1993), a ES não muda a relação de precedência entre nenhuma das atividades de projeto. Os benefícios vêm em função do aumento da comunicação entre todos os setores, redução de iterações, e melhoria do aprendizado de processos. Para sua completa adoção, talvez tenha faltado a noção de que o número de iterações não tende a diminuir, mas sim a sua amplitude, em função da correção precoce de problemas potenciais, oriunda do envolvimento de outras áreas de conhecimento.

Para GARDINER (1996), SMITH e REINERTSEN (1991) e SMITH (1997a) a co-locação (criação de um ambiente único para o núcleo da equipe) seja uma das maneiras mais eficazes para facilitar a integração de informações. HULL, COLLINS e LIKER (1996) enfatizam a importância da co-locação de equipes e a alteração dos critérios de gratificação. Segundo eles, equipes interdisciplinares trazem

uma série de disciplinas em contato, a co-locação facilita seu intercâmbio, e as gratificações por equipe fornecem incentivo para o trabalho colaborativo.

Nesta mesma linha, RAFII e PERKINS (1995) observam que comportamento, normas, estruturas sociais e sistemas de gratificação das organizações geralmente não suportam este processo. A ausência de comprometimento mútuo e confiança por todos os grupos pode obstruir ações individuais, fazendo com que os participantes não queiram compartilhar dados incompletos e informações preliminares, porque têm receio de críticas e são relutantes em executar retrabalhos para se adaptarem às exigências de outros. A comunicação informal, necessária para o desenvolvimento simultâneo, pode ser difícil de atingir num ambiente corporativo rígido, que se baseia em hierarquias sociais e de subordinação formais.

Grupos dispersos geograficamente aumentam os obstáculos para o sucesso da ES com equipes de desenvolvimento co-locadas. Distância física, além de diferenças de fuso horário e calendário de trabalho geram padrões complexos de comunicação e redes de interação que necessitam significativamente de mais coordenação. Além disso, quando a composição das equipes ultrapassa as fronteiras de um país, as diferenças sociais e culturais criam um grande potencial de conflitos e problemas de interpretação, adicionando outra camada de complexidade.

Segundo GOLDENSE (1997a), as equipes interdisciplinares devem ser coordenadas, preferencialmente, pelo representante da área mais comprometida com o produto. Se o produto for tecnologicamente inovador, deve-se optar por alguém da Engenharia. Se for algo cujo sucesso dependa de seu posicionamento estratégico no mercado, poderia ser alguém da área de *marketing*. Se o sucesso do produto depender de ajustes nos custos do mesmo e de seu desenvolvimento, poderia ser alguém do setor financeiro. Enfim, não há restrições e sim recomendações de forma que os principais pontos que definam o resultado do trabalho já sejam abordados na definição dos elementos chave da equipe (SMITH, 1997a).

2.6.2 Controle

A segunda abordagem na ES trata dos aspectos de controle na condução das atividades. Nesta, a ênfase é dada à captura e utilização de métricas, que são parâmetros levantados pelas gerências ou coordenações no sentido de nortear as decisões dos grupos de desenvolvimento.

A adoção de métricas para o controle de projetos pressupõe que o desenvolvimento de produtos seja, antes de tudo, considerado um processo, o que significa que seja realizável em fases de prazo e escopo estimados *a priori*. Observa-se ainda hoje uma certa dificuldade no sentido de compreender o desenvolvimento de produtos como uma atividade de condução previsível, de forma que possa ser planejada e cujos riscos possam ser estimados. Uma vez considerado como processo, este é passível de controle e automatização, comparável com um fluxograma de atividades (DOCHERTY, 1998).

KAPPEL e RUBENSTEIN (1999) observam que automatizar o processo do projeto de produtos requer que um algoritmo seja capaz de descrever e ajudar a executar uma tarefa estruturada. Entretanto, muito da prática de projetar não é analítica, e a maior parte dos projetistas não são peritos em programação matemática e análise. Além disso, os requisitos de projeto e metas de desempenho utilizados pelas técnicas de projeto geralmente contêm incertezas, e algumas delas podem nem sequer ser especificadas no início de um projeto.

Por outro lado, os mesmos autores alertam para os perigos do enquadramento de um trabalho criativo como o desenvolvimento de produtos sob a forma de processo, tal como o de fabricação em massa, sujeito a controles rígidos. Segundo eles, não se pode esperar que ambientes de desenvolvimento sem desafios produzam produtividade ótima ou criatividade. A resposta parece residir em alguma forma de equilíbrio.

RAFII e PERKINS (1995) indicam a importância da adoção de métricas como determinantes para o sucesso de projetos, ao afirmarem que quando o desenvolvimento distribuído e simultâneo faz sentido, os gerentes devem se concentrar

em fatores-chave como a liderança em projetos, estrutura organizacional e sistemas de métricas, porque estes têm um papel fundamental no resultado final dos projetos. HULL, COLLINS e LIKER (1996) propõem a adoção de critérios alternativos para a medição do sucesso de um produto, comparativamente às características tecnológicas inovadoras do mesmo. Segundo HULL et al. a ES representa uma mudança substancial para muitas corporações. Anteriormente, grandes empresas competiam introduzindo produtos inovadores, geralmente baseados em tecnologias e ciência radicalmente novos, oriundos de laboratórios de P&D. Enquanto muitos de seus produtos foram enormemente bem sucedidos no mercado, muitos falharam miseravelmente devido à inclusão de características novas que os clientes não desejavam, uma janela de oportunidade não aproveitada no mercado, ou altos preços. O sucesso requer que haja um equilíbrio entre a ênfase à inovação do produto e outros critérios de desempenho, como *time-to-market* e custos.

A expressão **controle de projeto** é definida por HULL, COLLINS e LIKER (1996), determinando seu escopo e traçando uma analogia com os mecanismos conhecidos de controle da produção. Controles sobre o projeto em processo referem-se a metodologias e protocolos comuns de projeto empregados pelos participantes. Isto inclui a utilização de padrões de projeto e revisões, que historicamente caracterizam as burocracias automatizadas para o monitoramento e controle de parâmetros como tempo e orçamento. Estes controles são análogos aos utilizados em fabricação, mas estendidos de maneira a incorporar o processo de desenvolvimento. Entretanto, assim como a fabricação tem crescentemente enfatizado a inspeção ao longo do processo ao invés de inspeções finais, os controles sobre o projeto são uma forma mais dinâmica de controle.

HULL e seus colegas afirmam ainda que uma diferença entre estes novos controles sobre projetos em andamento e os antigos controles comportamentais é que a responsabilidade pela concepção e execução de um projeto é grandemente descentralizada. Os controles são parcialmente estabelecidos por equipes de desenvolvimento semi-autônomas, cujo poder é aumentado dentro de limites para

melhorar o projeto de produtos de acordo com sua própria iniciativa. Esta maneira mais suave e dinâmica de controle pode ser relativamente mais eficaz para projetos complexos do que os controles rígidos e severos para se manter projetos dentro de seus cronogramas e orçamentos.

Paradoxalmente, controles sobre o projeto em andamento fornecem um complemento sinergístico à criatividade estimulada pelo envolvimento simultâneo e antecipado de outras áreas de conhecimento, porque o processo de desenvolvimento é enquadrado e mantido dentro de limites para aproveitar as oportunidades de mercado. O progresso de um projeto deve ser monitorado de acordo com o plano estabelecido e controlado utilizando estratégias predefinidas. Os chamados ***gates*** (marcos ou *milestones*) são relativos a eventos predefinidos em cada fase para controle do projeto. Se certas condições não forem satisfeitas, o projeto não continuará para o próximo estágio. Dois dos métodos para medir projetos em progresso são **Percentagem Completa** e **Valor Agregado**. A capacidade de controlar um projeto é diretamente dependente de monitoramento contínuo. De forma geral, os projetos são insuficientemente monitorados e as medidas de controle são tomadas tarde demais. Isto significa que a influência do gerente de projeto em custo, recursos e tempo é drasticamente reduzida.

HÁ e PORTEUS (1995) indicam que, do ponto de vista quantitativo, a otimização do espaçamento entre reuniões de revisão não traz tanto benefício quanto se possa imaginar. Um número reduzido de revisões igualmente espaçadas pode alcançar bons resultados. Do ponto de vista prático, muito do tempo de compressão, que é possível pela ES, pode ser alcançado introduzindo um pequeno número de revisões. Por exemplo, se dez revisões igualmente espaçadas forem conduzidas, ao invés de vinte e oito em espaços otimizados de tempo desiguais, o tempo previsto para conclusão é apenas 8,7% inferior ao ótimo, e produz acima de 90% da redução de tempo possível.

DOMAZET et al. (1995) sugerem que o processo de desenvolvimento possa ser avaliado através da evolução da base de dados de um projeto. Segundo eles, o

estado de um processo de desenvolvimento de produto pode ser mais bem capturado a partir de uma base compartilhada de dados do produto através de todos os participantes do processo. Ela contém dados do produto armazenados como resultado de suas atividades e pode refletir melhor o status do processo de desenvolvimento.

2.6.3 Ferramentas

A terceira e última abordagem na ES é a da aplicação da tecnologia da informação com o objetivo de promover a integração e facilitar o gerenciamento das atividades (MAHER e RUTHERFORD, 1997). Tal abordagem prevê a utilização desde ferramentas CAD até sistemas de trabalho colaborativo, ditas ferramentas CSCW.

Ferramentas CAD são ferramentas cujo objetivo é utilizar computadores com características de alto poder de desempenho numérico e gráfico (denominadas *workstations*) para o projeto de produtos. Tais sistemas foram desenvolvidos a partir da década de setenta, mas foram definitivamente implantados nas empresas de projetos na década seguinte, acompanhando a redução dos custos dos computadores, *software* e dispositivos de rede. Hoje existem inúmeras plataformas comerciais disponíveis no mercado de fabricantes como IBM, PTC, Autodesk, Mentor Graphics, entre outros. Sua integração com ferramentas CAE, que utilizam métodos numéricos para análise (estrutural, térmica, compatibilidade eletromagnética e outras) e CAM, que permitem a geração de dados para fabricação através de máquinas de comando numérico, tem melhorado significativamente nos últimos anos. Porém, ainda há consórcios internacionais para o desenvolvimento de formatos de dados intercambiáveis entre estas plataformas (ROSCHE, 2003).

Ferramentas do tipo PDM são ferramentas que funcionam como bancos de dados de projeto, com funcionalidade tal que permite o compartilhamento de informações, mantendo a integridade das mesmas mediante dispositivos de versionamento automático e garantindo a consistência dos arquivos de dados na

medida em que permitem a vinculação de arquivos de diversas modalidades com níveis de abstração (aspectos do projeto) distintos e mesmo entre plataformas diferentes, como planilhas, textos, desenhos, modelos geométricos, esquemas elétricos, entre outros (BAILEY e RUCKER , 1998).

Ferramentas de *workflow* são programas computacionais que permitem a modelagem e controle de processos definidos em tarefas interdependentes. Podem ser imaginadas como uma evolução dos antigos fluxogramas, desenhos que descreviam a funcionalidade de programas computacionais (DANGELMAIER, KRESS e WENKI, 1999). Tais ferramentas facilitam a identificação e o monitoramento contínuo da situação das atividades em andamento, de forma que seja possível a introdução de métricas de controle (CRUZ , 1998; THRONICKE e SCHOLZ, 1998).

Ferramentas CSCW são sistemas computacionais que permitem o trabalho colaborativo e simultâneo entre participantes de uma equipe. Para tanto, disponibilizam aos usuários um ambiente de troca de informações mais rico e sofisticado que as ferramentas comuns de correio eletrônico, pois oferecem funcionalidades assíncronas avançadas como controle de acesso a documentos eletrônicos (direitos de alteração, consulta e eliminação), informações qualitativas sobre a consulta ao sistema, versionamento, entre outras. Além disso, alguns sistemas permitem a colaboração síncrona, ou seja, simultânea, através de dispositivos de áudio e vídeo conferência e *markups* (anotações) eletrônicos (PALMER e FIELDS, 1994). São exemplos de ferramentas CSCW amplamente difundidas: Microsoft Netmeeting (ELLIOT e LASSER, 2001), IBM Lotus Notes e BSCW.

KAPPEL e RUBENSTEIN (1999) indicam que as ferramentas a serem adotadas com o propósito de facilitar o trabalho de projetistas devam levar em conta os aspectos sociais do trabalho de desenvolvimento em equipes. As incertezas, influências organizacionais, e experiências de projetistas combinam para formar o ambiente de desenvolvimento de um projeto. O ambiente não pode ser separado do problema quando novas tecnologias em projeto forem implementadas. O potencial da tecnologia da informação está em assistir grupos com melhoramento criativo real sem

sacrificar a infra-estrutura social existente na organização. Sugerem ainda que exista um problema fundamental a respeito da organização social para o projeto, o qual, quando embutido num programa computacional inflexível e implementado nas organizações, pode causar resistência, comportamento caótico e resultados ruins. Mais tecnologias úteis poderiam surgir a partir de uma melhor compreensão do processo de comunicação no projeto de produtos, que é natural e visualmente orientado.

Em seu trabalho, KAPPEL e RUBENSTEIN (1999) defendem a adoção de tecnologia da informação, porém demonstram certo ceticismo ao considerar os resultados práticos que têm sido obtidos no aspecto da sua adoção em favor da criatividade do projetista. Para eles, o potencial da tecnologia da informação no projeto de produtos não está ligado diretamente à sua existência, mas depende sim de sua implementação e uso continuado na indústria. A criatividade não tem se beneficiado tanto da introdução de tecnologia da informação quanto a coordenação. CAD é um exemplo em que os interesses da coordenação e eficiência aparentemente dominaram os interesses da criatividade e trabalho conceitual.

RAFII e PERKINS (1995) incentivam a adoção de ferramentas cujo retorno não pode ser sentido em curto prazo, e cuja contabilização de custos deve ser revista. As barreiras para as atividades de ES poderiam ser reduzidas pela modificação dos sistemas de contabilização de custos a fim de evitar penalizar grupos por investimentos esporádicos em sistemas de informação, cujo retorno é de longo prazo. Por outro lado, SMITH (1997a) lança a hipótese de que o investimento em tecnologia da informação pode não ser o foco principal de quem procura implementar os conceitos de ES. Segundo o pesquisador, tecnologias computacionais e de comunicação não são necessárias para a adoção de práticas de ES. SMITH sugere que as novas tecnologias não são nem necessárias, nem suficientes para a cooperação interdisciplinar, mas servem para reduzir as barreiras contra a adoção de tal cooperação e têm estimulado os tipos de comunicação e análise que ocorrem no processo de desenvolvimento de produtos.

A respeito dos benefícios da adoção de tecnologia da informação no DIP,

HULL, COLLINS e LIKER (1996) indicam que a tecnologia da informação e computação têm um papel vital em ajudar equipes interdisciplinares a atingirem altos níveis de desempenho. Por exemplo, bases de dados *on-line* permitem que opções de projeto tecnicamente sofisticadas sejam simultaneamente avaliadas por participantes de várias especialidades, incluindo fornecedores e funções posteriores como *marketing* e clientes.

GARDINER (1996) sugere que os sistemas de comunicação na abordagem de ES devem, de maneira ideal, fornecer um ambiente onde a co-locação de membros da equipe seja desnecessária, em função da riqueza das comunicações fornecerem condições similares às de uma reunião face-a-face. As condições podem até ser melhores, porque o processo de se chegar a uma decisão de projeto pode ser capturado automaticamente pelo sistema. A infra-estrutura de comunicação deve suportar o fluxo de trabalho do projeto, a tal ponto de controlar o cronograma, identificar omissões durante a sua realização e permitir a aprovação da documentação através da rede.

Por outro lado, KING e MAJCHRZAK (1996) apontam grandes deficiências nas ferramentas existentes para o apoio a ambientes de ES ao destacar que há diversas suposições tomadas por certas pelos desenvolvedores de sistemas, as quais não estão alinhadas com o comportamento humano constatado perante a tecnologia da informação e também levando-se em consideração o processo social intrínseco ao desenvolvimento de produtos. Os autores identificam nove suposições e propõem sua reformulação baseando-se tanto nestes fatores quanto na experiência acumulada no desenvolvimento e implementação de outros sistemas informatizados no passado recente. KING e MAJCHRZAK sugerem estratégias para o desenvolvimento de ferramentas de ES que sejam centradas no usuário, disponham de mecanismos que capturem as informações nas realidades contextualizadas dos participantes da equipe de desenvolvimento e que sejam adequadas às diferentes etapas de criação e projeto de produto.

2.6.4 Gestão

As três abordagens mencionadas são permeadas pelo gerenciamento, uma função essencial para a implementação de um ambiente de ES. BLACKBURN et al. (1996) e BLACKBURN, HOEDEMAKER e WASSENHOVE (1996) apresentaram resultados de uma investigação referente às práticas de gerenciamento de projetos de *software*. Esta área em particular sofre em função de vários fatores que afetam o seu nível de maturidade, uma vez entendido como processo, comparativamente ao desenvolvimento de artefatos. Entretanto, muitos problemas são comuns e ficam até mais acentuados pelo nível intelectual que marca a realização destes trabalhos. Os autores indicam que a maneira pela qual o tempo é alocado ao longo das etapas do projeto tem um importante efeito sobre o desenvolvimento de *software*. A maioria das empresas de *software* dedica pouco tempo e recursos para as etapas preliminares do projeto. Tempo e esforço crescem à medida que o projeto evolui, do conceito até codificação e testes. Concluem que a peça faltante no quebra-cabeças seja o gerenciamento. Segundo eles, para alcançar seu potencial em desenvolvimento de *software*, a ES deve ser praticada através de um esforço coordenado de gerenciamento por todas as etapas do projeto (e entre projetos) ao invés de ser uma aplicação selecionada de ferramentas.

BLACKBURN et al. enfatizam também a questão de reuso. Para eles, as firmas não possuem uma estratégia pro-ativa, de projeto para o reuso; elas não têm sucesso ao lançar programas em torno de objetos reutilizáveis, sendo que a maior parte dos engenheiros de *software* prefere não reutilizar código. O maior incentivo para o reuso, neste caso, tem sido não oferecer aos programadores tempo suficiente para desenvolver código novo.

BLACKBURN e seus colegas criticam algumas práticas comuns para o desenvolvimento de produtos, particularmente de *software*, decorrentes talvez da natureza humana no gerenciamento e falta de planejamento. Segundo eles, à medida que um projeto evolui por suas etapas desde os requisitos do cliente até testes, os

recursos alocados ao projeto aumentam de forma estável. Concluem que os cínicos podem sugerir que isto seja típico de gerentes de projeto: eles reagem à pressão de tempo através do aumento de recursos na esperança de que este tipo de postura encurtará o tempo de desenvolvimento. Infelizmente, tal comportamento é comum em muitos novos esforços de desenvolvimento, particularmente no caso de *software*. Há indícios de uma forte correlação entre velocidade de desenvolvimento de *software* e tempo empregado na etapa de especificação do processo.

Um tema presente na literatura sobre desenvolvimento de novos produtos refere-se ao fato de que as empresas gastam muito pouco tempo identificando as necessidades do cliente e que mais tempo empregado nesta parte do processo poderia levar a reduções no tempo total de desenvolvimento. BLACKBURN e seus colegas evidenciam os efeitos do não envolvimento dos clientes nas etapas de levantamento de requisitos e, possivelmente, planejamento deficiente. Eles sugerem que os gerentes são constantemente frustrados pela alteração de requisitos - em todas as fases do projeto e também nos vários setores envolvidos - e que o problema aumenta quando há projetos de interface entre *hardware* e *software*. Ainda sobre esta questão, SMITH (1997b) indica que o fato de não responder às necessidades do mercado é uma falha contínua em projeto e desenvolvimento de produtos, e tem sido reconhecida como tal há muito tempo. Por sua vez BLACKBURN e seus colegas sustentam a idéia de que através da utilização de mais tempo para elaborar as corretas especificações, uma economia substancial de tempo pode ser alcançada no projeto e testes, pela eliminação de grande parte do retrabalho causado por requisitos do cliente mal interpretados.

Ainda sobre o levantamento de necessidades dos clientes, BLACKBURN e seus colegas relatam a respeito da pouca utilização prática de técnicas de QFD para o desenvolvimento de *software*. Segundo eles, apesar de muito ter sido escrito a respeito de QFD (CROW, 1996), há pouca evidência da utilização desta ferramenta, concebida para captar os requisitos do cliente e convertê-los em especificações de produto para *software*. Segundo BLACKBURN e seus colegas, QFD é muito mais dominante no Japão, pouco visível nos Estados Unidos e virtualmente desconhecida na Europa.

A dificuldade de coordenação do trabalho interdisciplinar em ambientes funcionalmente hierarquizados destacada por CRABTREE, FOX e BAID (1997) e SMITH (1997a). Segundo estas pesquisas, há várias considerações que podem ter desencorajado os tipos de cooperação interdisciplinar associados à ES em organizações hierarquizadas e funcionais. Em primeiro lugar, em uma organização que estimule o trabalho interdisciplinar, as linhas de controle podem se enfraquecer. Em segundo lugar, mais trabalho de comunicação e coordenação se faz necessário, o que tem o potencial de distrair do trabalho fundamental (técnico) dos empregados. Estas dificuldades podem levar automaticamente à não adoção de uma abordagem de ES. E conclui que a característica mais perigosa da cooperação é a tendência de utilizá-la como uma saída fácil para as dificuldades que necessitam de habilidade gerencial para superá-las. SMITH sugere que haja vantagens em organizações tipicamente funcionais, que são conseguidas a um determinado preço, pois organizações funcionais são melhores para reter conhecimento avançado em áreas técnicas específicas. Este conhecimento tipicamente surgiria às custas de coordenação interdisciplinar e *lead time* no desenvolvimento de produtos.

O papel da alta gerência é fundamental para a implementação da ES. TIBBITTS e KEANE (1995) se referem ao gerenciamento como a **força motriz** por trás da implementação da ES. Devem, segundo os pesquisadores, exercer um papel pro-ativo e de liderança em efetivar tais mudanças se as organizações perceberem os benefícios da mesma. TANG, JONES e FORRESTER (1997) vão além, e definem que a ES não seja um processo físico ou um conjunto de procedimentos; é, sim, a responsabilidade em nível gerencial de integrar as equipes em unidades de negócio coesas e eficazes. Há também de se colocar a importância do aprendizado em cada projeto como vantagem competitiva. Se os funcionários forem ensinados que cada projeto tem duas dimensões - que o que importa não é apenas o produto ou processo resultante, mas **como** o resultado foi alcançado - eles assimilarão a idéia de que o aprendizado é o objetivo primário para todos na organização.

BLACKBURN, HOEDEMAKER e WASSENHOVE (1996), na tentativa de

desencorajarem a adoção de ferramentas de suporte gerencial como solução para os problemas de cunho administrativo, recomendam que a chave para o desenvolvimento de *software* de qualidade reside no processo e como o mesmo é gerenciado. Ferramentas são importantes, mas o processo é transcendente. Para os autores, a fim de reduzir tempo de desenvolvimento, gerentes de equipes de desenvolvimento de *software* atualmente obtêm melhores resultados através do gerenciamento de pessoas e de processos interdisciplinares do que através do uso de ferramentas CASE e tecnologia.

A respeito da estrutura organizacional que suporta a adoção dos princípios da ES, RAFII e PERKINS (1995) destacam a necessidade de mudanças organizacionais para permitir a troca de informações. Para eles, a comunicação interativa, em dois sentidos, necessária para a ES eficaz, precisa de uma organização horizontalizada com tomada de decisão informal e direta e trocas de informação em níveis mais baixos. Para ser ágil, a informação deve fluir horizontalmente, sem viagens verticais freqüentes para aprovação pela gerência.

SUSMAN e DEAN (1992) reforçam esta idéia de mudança organizacional ao proporem o conceito de *empowerment* ou fortalecimento dos gerentes de projeto e alterações nos sistemas de gratificação. Segundo estes pesquisadores, para ser bem sucedida, a ES também requer mudanças de suporte no ambiente organizacional que envolve uma equipe. Há, por exemplo, a necessidade de transferir a autoridade da tomada de decisões dos gerentes funcionais para as equipes e reformular o sistema organizacional de gratificações para encorajar o comportamento orientado a projetos.

Por sua vez, TANG, JONES e FORRESTER (1997) sugerem que se uma organização quiser adotar ES integralmente, sua estrutura organizacional e cultura devem mudar. Em alguns casos pode ser suficiente instruir Engenharia do Produto, Qualidade e Engenharia de Produção de que eles devem repensar a maneira pela qual desenvolvem produtos. Entretanto, é necessário mudar também a estrutura da organização.

O desenvolvimento de produtos é na verdade um processo muito dinâmico.

Por isso DOMAZET et al. (1995) colocam claramente o papel decisivo da coordenação na mediação de conflitos. Isto advém da ocorrência de muitos eventos excepcionais e das dificuldades em antecipar que a decisão tomada por um especialista, ao reagir a um evento, conflita com outros e, então, precisa ser coordenada. RAFII e PERKINS (1995) reforçam esta idéia ao afirmarem que, apesar de liderança ser importante para todos os projetos, um gerente forte é absolutamente vital para um programa de desenvolvimento simultâneo distribuído.

HANDFIELD (1994) procurou investigar mais a fundo as alterações no gerenciamento provocadas pela mudança de padrões de comunicação entre membros das equipes de desenvolvimento. Para ele as oportunidades para sobreposição de tarefas podem ser criadas ao considerar os tipos de dados que devem ser transmitidos de uma atividade para outra, para fornecer subsídios à tomada de decisões. Ao compartilhar informações “imperfeitas” em vários pontos, as tarefas seguintes podem ser iniciadas antes das predecessoras terem sido completadas Este é um fenômeno conhecido como **Lei de Brookes**. Para implementar cronogramas paralelos, mudanças profundas nos estilos de tomada de decisão são necessárias. Não obstante, o tempo total necessário para o desenvolvimento pode ser reduzido substancialmente através do desenvolvimento em paralelo.

2.7 VANTAGENS E DESVANTAGENS DA ENGENHARIA SIMULTÂNEA

É comum encontrar-se nas referências sobre ES posicionamentos totalmente favoráveis sobre seus conceitos e aplicabilidade. Autores com PRASAD (1996a), sugerem dados quantitativos em termos de ganho em tempo ou custos, levantados junto a empresas que implementaram esta abordagem. Entretanto, também há autores, como AITSAHLIA, JOHNSON e WILL (1995), e CRAWFORD (1992), que enfatizam aspectos negativos destas práticas ou pelo menos a omissão de certos aspectos relevantes para a sua implementação.

BLACKBURN, HOEDEMAKER e WASSENHOVE (1996) descrevem a

sistemática de desenvolvimento de produtos através dos preceitos da ES enfatizando que haja duas formas pelas quais o desenvolvimento de produtos se beneficia da sobreposição de atividades. Primeiro, atividades paralelas são estimuladas. Com a liberação antecipada de informação, os engenheiros podem começar a trabalhar em diferentes fases do problema, enquanto o projeto final continua evoluindo. Isto não só torna possível a ES, como também encoraja atividades simultâneas entre projeto, Engenharia e teste. A equipe de desenvolvimento trabalha menos como uma equipe de revezamento (passando o bastão após cada volta para o próximo membro da equipe) e mais como uma equipe de retaguarda no automobilismo, onde todos trabalham simultaneamente em busca de um único objetivo. Por fim, segundo os autores, o retrabalho que consome tempo é evitado na medida em que a liberação antecipada de informações promove a detecção antecipada de discrepâncias em relação às metas programadas. Correções intermediárias no projeto, apesar de acrescentarem tempo nas fases iniciais, tendem a produzir tempos totais menores. A alternativa de revisar projetos acabados, e descobrir que fogem dos objetivos previstos, geralmente leva a várias iterações completas do ciclo de projeto e tempos totais maiores.

A correlação entre a prática da ES e bons resultados no desenvolvimento de produtos nem sempre é clara entre os pesquisadores. SMITH (1997b) reconhece os benefícios da ES indicando que haja muita similaridade entre o conceito de ES e as melhores práticas em desenvolvimento de produtos. Ao mesmo tempo, procura justificar o motivo pelo qual não foi imediatamente adotada em muitas empresas. Para SMITH, isto ocorreu tanto porque métodos mais antigos pareciam mais fáceis, quanto porque o sistema educacional não defendeu suficientemente que uma mudança deveria ser promovida. Ainda segundo SMITH, educadores e pesquisadores têm a obrigação de assegurar que isto não venha a ocorrer novamente, sugerindo que permaneçam atentos à evolução das melhores práticas para o desenvolvimento de produtos.

Para RAFII e PERKINS (1995) a validade do modelo de ES para produtos tipicamente inovadores é questionável. Segundo eles, os custos e riscos da ES são função da incerteza tecnológica, um atributo de projetos tipicamente relacionado com

suas fases no ciclo de vida. Projetos que lançam produtos inovadores tendem a ter altos níveis de incerteza ou complexidade. Para produtos inovadores, o desenvolvimento tradicional seria apropriado porque os riscos do desenvolvimento em paralelo ultrapassam os ganhos em potencial. Na mesma linha, AITSAHLIA, JOHNSON, e WILL (1995) reforçam a idéia de que, apesar de disseminadas na indústria, as idéias de paralelismo de atividades preconizadas pela ES possam não ser válidas para qualquer produto. Para eles, a incerteza no processo de desenvolvimento de produtos provoca a erosão dos benefícios da ES, e em alguns casos, um projeto desenvolvido com tarefas em paralelo pode ser pior que em série.

A suspeita de que os conceitos da ES podem não ser benéficos para todos é levantada também por HANDFIELD (1994), que diferencia resultados obtidos para produtos **incrementais** ou também denominados **derivados**. Segundo ele, inovações incrementais são mais apropriadas para a abordagem de ES, em oposição à abordagem tradicional, e podem ainda ajudar a melhorar a qualidade do produto.

Por outro lado, HULL, COLLINS, e LIKER (1996) destacam as diferenças de resultado considerando os diferentes sistemas de produção, em função do tipo de produto a ser desenvolvido, levantando a hipótese de que a ES possa ter um impacto bem maior no desempenho de sistemas de produção em massa do que por lote. CRAWFORD (1992) a ES tem uma “etiqueta de preço muito maior do que o que comenta a maioria dos seus eufóricos e muito ocupados praticantes”. Para CRAWFORD, os dados comumente referenciados a respeito dos prejuízos causados mais por atrasos do que por custos excessivos se aplicam somente a mercados de crescimento acentuado e produtos cujos ciclos de desenvolvimento são reduzidos (ver Figura 2). O desenvolvimento acelerado de produtos combinaria bem com linhas de produtos que têm ciclos de vida curtos, baixos custos de cancelamento, facilidade de entendimento dos requisitos do cliente sobre novas características e barreiras reduzidas para sua entrada no mercado.

EVBUOMWAN e ANUMBA (1996) enumeram as seguintes vantagens na adoção da ES como sistemática de trabalho em oposição à sistemática seqüencial

tradicional:

- segregação, isolamento e descomprometimento com o sucesso do produto são virtualmente removidas da empresa, de forma que os diversos departamentos podem trabalhar em conjunto e de forma integrada;
- o processo global de desenvolvimento do produto é reduzido à medida que as etapas são realizadas em paralelo;
- a capacidade é incrementada para a competição global e possibilidade de fornecimento em tempo de produtos de alta qualidade e custos reduzidos aos clientes;
- há menos erros e falhas de projeto, assim como número reduzido de alterações de projeto;
- as revisões de projeto e iterações no desenvolvimento do produto são reduzidas e possivelmente eliminadas;
- ocorre melhoria da comunicação e cooperação entre projetistas, gerentes e outros profissionais envolvidos no processo de desenvolvimento do produto;
- há maior envolvimento dos funcionários dentro da organização;
- novos produtos podem ser lançados no mercado com maior grau de satisfação dos clientes, menores custos e maior nível de qualidade;
- as empresas podem ser mais solícitas aos clientes.

Em síntese, DOWLATSHAHI (1994) menciona dois benefícios evidentes no uso da ES: redução no *lead time* de desenvolvimento de produtos e economia geral de custos.

TIBBITTS e KEANE (1995) relacionam os seguintes benefícios:

- desenvolvimento acelerado de produtos;
- fabricação mais eficiente em termos de custo;
- melhores projetos que beneficiam clientes;
- melhoria de qualidade e confiabilidade;

- melhores produtos que satisfazem usuários finais e
- melhores controles de custo.

HULL, COLLINS, e LIKER (1996) procuram enfatizar a importância de redução dos custos provocados pela perda de oportunidades de mercado. Segundo estes pesquisadores, empresas que implementam ES reduzem os custos porque melhores decisões de projeto são tomadas em etapas preliminares. Isto é importante porque a grande maioria dos custos posteriores é de certa forma definida na fase conceitual, e as alterações são exponencialmente mais caras que em etapas anteriores. A ES também reduz custos de oportunidade através da compressão de tempo. Custos de oportunidade ocorrem quando os produtos são tecnicamente um sucesso, mas pecam nos ambientes de fabricação ou *marketing*. Estes custos de oportunidade podem ser enormes porque as empresas que chegam atrasadas ao mercado geralmente perdem porções significativas de sua base de clientes potenciais.

2.8 TECNOLOGIA DA INFORMAÇÃO

2.8.1 Modelos de Informação

Segundo PRASAD (1996a), há cinco classes que representam os modelos de informação em produtos mecânicos e sistemas para ES, ilustrados na Figura 4, quais sejam:

- modelos corporativos;
- modelos de especificação ou requisitos;
- modelos de produto;
- modelos de processo e
- modelos cognitivos.

Os modelos corporativos fornecem uma descrição precisa das atividades, conectividade de dados e rede de comunicações de uma empresa. Através de ações documentadas em fluxogramas, a empresa captura o fluxo de informações através das

unidades envolvidas no projeto e desenvolvimento de um produto. O modelo reflete as necessidades de informação tanto de unidades que operam de forma automatizada quanto não automatizada, e consiste de quatro bases de dados:

- Dados de engenharia,
- Dados de manufatura,
- Dados de controle de manufatura e
- Dados de cronogramas mestres.

Especificação é a organização ou alocação das informações que chegam em uma série de entradas, necessidades e restrições. Uma restrição é uma limitação a respeito do ambiente a que o artefato será submetido. Os elementos da especificação fazem o mapeamento do fluxo de informações que partem de peças, processos de fabricação, planejamento de processos de fabricação, biblioteca de ferramentas, cronograma de produção para diferentes planos: manufatura, montagem e inspeção. Os elementos trabalham em processos e técnicas para se obter e traduzir as vontades do consumidor em produzir hierarquicamente um conjunto de objetivos descritos por especificações funcionais e modelos.

A classe de modelos de especificação fornece uma documentação para o planejamento dos passos para o desenvolvimento de um produto, o qual a empresa seguirá na prática. O processo começa com uma abordagem *top-down*. Por exemplo, especificações de sistema serão primeiramente destacadas e então detalhadas em passos seguintes, gerando um grupo telescópico de especificações. Este processo é repetido ao se aumentar o grau de detalhamento até que o nível inferior da estruturação seja atingido.

A classe de modelos de informação que menos dispõe de alternativas quanto ao desenvolvimento de sistemas é a classe de modelos de especificação. É importante lembrar, contudo, que há sistemas que fazem uso de ferramentas para a captura de necessidades de clientes a fim de convertê-las em especificações. É o caso de ferramentas de QFD (CROW, 2003), que, se utilizadas apropriadamente, podem auxiliar na tarefa de mapear as informações de forma que sejam integráveis aos demais

modelos de informação.

Segundo PRASAD, a terceira classe de modelos é a de modelos de produto, cuja principal finalidade é de fornecer informações ao usuário de forma não ambígua e completa. Modelos nesta classe estruturam informações a respeito das características do artefato em múltiplos níveis de abstração. Níveis de abstração são pontos de vista em relação ao produto, que procuram descrevê-lo conforme o nível de detalhamento desejado. Portanto, um esquema de funcionamento de um produto utilizando diagramas em blocos fornece um nível de abstração superior a um desenho contenho detalhes construtivos e medidas. Porém, ambos são, de alguma forma, componentes do mesmo modelo de produto. Isto, para segundo PRASAD, não é o mesmo que modelagem de dados do produto (PDM), sistema este que estrutura elementos de dados de uma forma comum ou uniforme, utilizando formatos de dados padronizados, como STEP ou IGES (ANSALDI e GIANNINI, 2000).

Entretanto, tal ponto de vista pode não estar de acordo com o que se observa no mercado de ferramentas de PDM disponível atualmente. Nestes sistemas há a possibilidade de se mesclar diferentes formatos de dados, e garantir alguma espécie de sincronização destas informações para manter a sua integridade. Por outro lado, a visão do produto através de informações separadas em nível de abstração não é contemplada na maioria dos sistemas atuais.

A classe de modelos de produto tem uma série de propósitos, dependendo dos objetivos das equipes de desenvolvimento. Por exemplo, equipes de projeto podem utilizar tais modelos para observarem e analisarem diferentes alternativas. Estes modelos capturam a intenção do produto numa perspectiva de ciclo de vida, considerando as necessidades do cliente, desempenho, geometria, carregamento e assim por diante, que afetam o comportamento do produto de alguma maneira. As áreas de *marketing* e vendas podem utilizá-lo com o objetivo de verificar sua aceitação e estimar preços, comparar com os produtos concorrentes ou propor um novo produto que ainda não tenha similar no mercado.

A quarta classe de modelos refere-se aos de processo, que têm a capacidade

de estruturar a informação de estados ou conhecimento do processo. De maneira típica, o processo necessário para se levar um projeto através do seu ciclo de vida não é muito bem entendido, especialmente em organizações complexas. Como os problemas são resolvidos, os métodos de criação ou modificação de um produto, o seqüenciamento de eventos, as questões de manufatura e o conjunto de informações coletadas formam uma grande porção deste universo de conhecimentos do processo. Além disso, o conhecimento contido em manuais de procedimentos, memorandos, livros textos, resultados de testes e experimentos, não estão bem integrados no sentido de projetar e desenvolver um produto complexo.

A última classe de modelos sugeridos por PRASAD refere-se aos modelos cognitivos. Os modelos anteriores descrevem o produto e comportamento do processo, mas o problema não é resolvido a menos que um cérebro humano ou, mais precisamente, um “cérebro virtual” aja sobre eles. O processo de formação de equipes (compartilhamento de mentes cognitivas) e compreensão uniforme não é bem entendido. A maioria dos gerentes considera a formação de equipes interdisciplinares uma questão de gerenciamento. Apesar do gerenciamento ser um grande catalisador na formação de equipes e motivação, a formação efetiva de equipes é um processo natural de interconexão humana informal e a criação de um profundo senso de compreensão mútua entre os membros. O modelo cognitivo captura o comportamento humano em equipes e situações de colaboração. Estrutura cognitiva é, portanto, a maneira como os membros de uma equipe pensam e agem durante a tomada de decisões. A maioria dos ambientes de ES envolvem um relacionamento cuidadosamente orquestrado entre equipes e máquinas. Para PRASAD os modelos de informação são incompletos sem uma estrutura comum para o entendimento cognitivo.

2.8.2 Sistemas de Informação

A aplicação de tecnologia da informação no ambiente industrial de hoje se constitui numa variedade enorme de sistemas espalhados pelos diversos setores das

corporações. Considerando-se por exemplo os sistemas para o tratamento da classe de modelos de produto, há inúmeras ferramentas que permitem a modelagem sólida, funcional, entre outras, que, por sua vez, podem ou não estar integradas. Cada uma destas ferramentas agrega informações ao modelo do produto, que evolui tal como um produto numa linha de montagem (PATTERSON, 1993). Entretanto, uma das maiores dificuldades, apesar da tecnologia de CAD/CAE/CAM estar disponível há décadas, é a perfeita integração entre elas de forma que não haja necessidade de retrabalhos ou falhas de interpretação. Existem diversas iniciativas no sentido de tornar a comunicação de informações dentro do modelo mais eficaz, o que demonstra que este assunto está longe de ser encerrado (PDM IMPLEMENTOR FORUM, 2003).

Com o advento das redes corporativas, facilitado por custos cada vez menores de equipamentos de informática e a massificação de seu uso, as empresas criaram sistemas que tratam individualmente de cada questão na corporação. Um dos problemas a serem resolvidos é a falta de integração entre sistemas de compras, inventário, vendas, Engenharia, entre outros, do ponto de vista de classes de modelos de corporação. Uma alternativa recente aos apelos da indústria para que se chegue a uma solução para este impasse tem sido a adoção de sistemas ERP. Paralelamente a estas implementações foi necessário que se desenvolvesse a Reengenharia dos Processos do Negócio (BRP) que redefine as rotas dos fluxos de informação. De qualquer forma, esta iniciativa, cujos resultados variam de empresa para empresa, abraçam apenas algumas das classes de modelos de informação descritos por PRASAD (1996a).

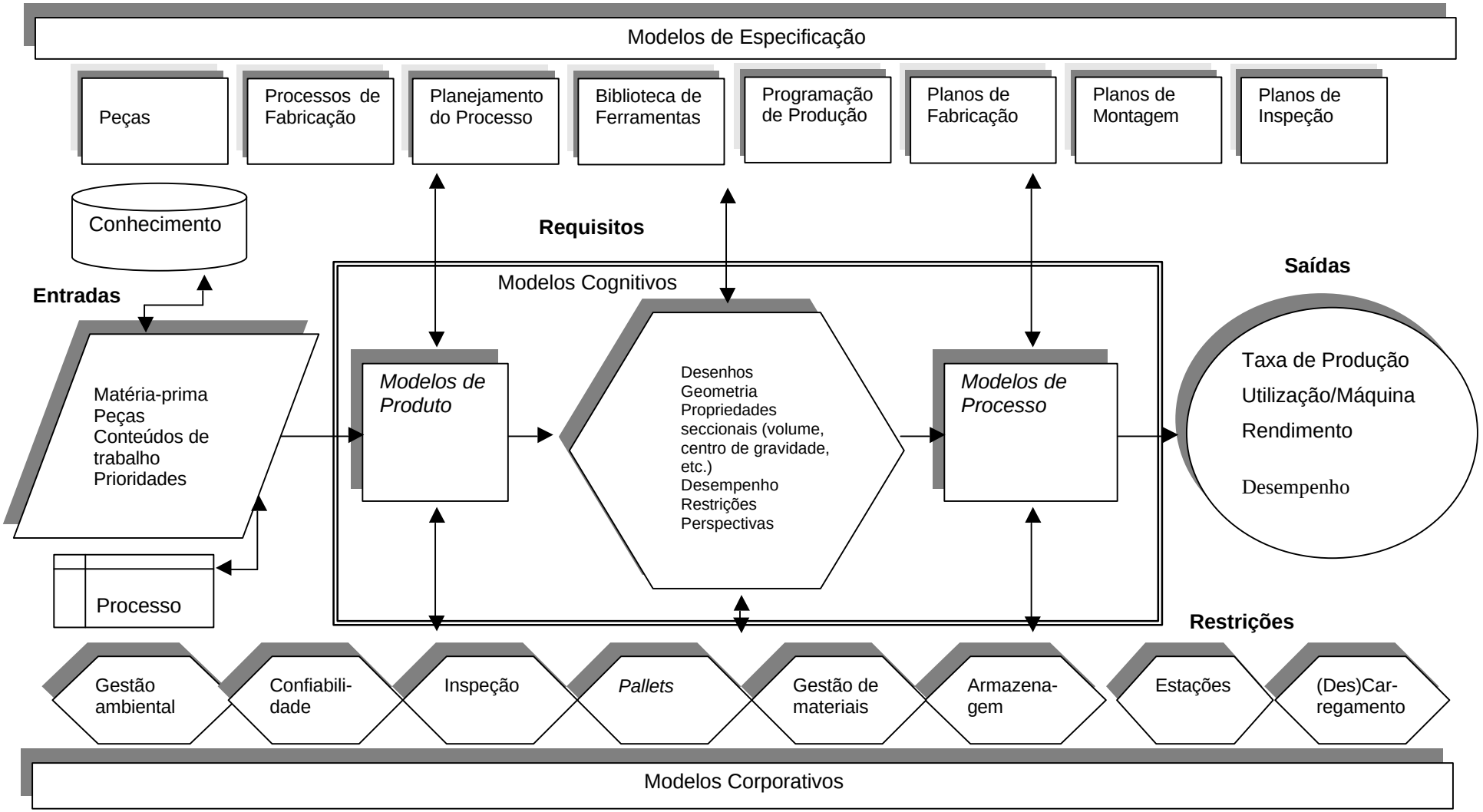
Com o foco em modelagem de processos, pode-se dizer que as ferramentas de CAPP, no ambiente da manufatura e *workflow,* no ambiente de gerenciamento e desenvolvimento de produtos, tendem a abordar o problema da identificação do ordenamento, seqüencial ou não, de tarefas. Tais ferramentas, especialmente as de *workflow,* podem ser tremendamente úteis para o controle e gerenciamento dos processos, uma vez que podem capturar o status das atividades e inserir listas de verificação de forma a garantir a qualidade das informações que serão trocadas ou

compartilhadas (CRUZ, 1998).

Segundo DARNTON (1995), uma outra categoria de sistemas de informação disponível hoje é aquela que abrange os chamados sistemas computacionais de suporte ao trabalho colaborativo (CSCW). Trata-se de sistemas que auxiliam pessoas que trabalham juntas em grupo para produzirem algum produto ou serviço. São, portanto, direcionados à classe de modelos cognitivos de informação, pois visam tratar dos fatores sociais envolvidos durante o trabalho de desenvolvimento de produtos, objetivando em última instância a tomada de decisões.

Com relação aos detalhes de implementação de sistemas de informação, percebe-se uma tendência clara atual no sentido de se utilizar padrões consagrados pela popularização do uso da *internet* no meio comercial, industrial e mesmo doméstico. Navegadores (*browsers*) podem ser utilizados como interface para sistemas que consultam bancos de dados relacionais ou objeto-orientados através de protocolos de comunicação como HTTP e FTP e adicionam ou extraem informações relevantes ao usuário. Além disso, como são interfaces utilizadas para diversos fins, seu aprendizado é bastante rápido pelos usuários. Portanto acredita-se que em pouco tempo não haverá mais espaço para aplicativos que não apresentem tais facilidades. Exemplo disso são os sistemas de PDM, que nasceram como sistemas que dispunham de interface proprietária, e hoje já se rendem aos padrões de interface do mercado baseados em navegadores.

Figura 4 - Modelos de informação



Fonte: Adaptado de PRASAD (1996a, p. 345).

2.8.3 Gerenciamento de Projetos

Uma das alternativas encontradas pelas empresas a partir da década de setenta para o aumento de sua competitividade, diante de um cenário de contínua redução nos ciclos de vida de produtos e aumento das exigências dos consumidores por produtos melhores, mais baratos e inovadores, tem sido a adoção de estruturas organizacionais que permitem a substituição do paradigma tradicional, baseado no gerenciamento informal de projetos, em favor de outra que favorece o controle e monitoramento de projetos (KERZNER, 1995). Entretanto, as novas estruturas passaram a coexistir com estruturas organizacionais tradicionais, funcionalmente distribuídas, que privilegiam o desenvolvimento do conhecimento específico em detrimento da integração entre funções. Os projetos, conduzidos de forma seqüencial, eram realizados de forma que os departamentos se responsabilizavam apenas pela sua área de conhecimento aplicada naquele projeto (PRASAD, 1996a). Tal abordagem tem sido substituída por outra em que há a presença de um gerente de projetos, que coordena os trabalhos de uma equipe de desenvolvimento interdisciplinar, cujos membros dividem uma responsabilidade que transcende a de suas áreas específicas de conhecimento, uma vez que seu sucesso depende do sucesso do produto no mercado.

A introdução da figura do gerente de projetos trouxe a necessidade de mudanças estruturais nas organizações, de forma que organogramas puramente funcionais foram substituídos por organogramas matriciais, em que os gerentes funcionais têm sido gradualmente transformados em gerentes de recursos. Cabe a estes a definição pela alocação de recursos humanos aos projetos em andamento, além, evidentemente, da alocação de instrumentos para que os trabalhos possam ser realizados. Há, na verdade, diferentes graus de implementação destas novas estruturas, nas quais variam os poderes atribuídos aos gerentes funcionais e de projeto. O gerente de projeto pode, em estruturas que tendem às antigas estruturas funcionais, se responsabilizar apenas pelo monitoramento dos cronogramas de desenvolvimento,

reservando aos coordenadores funcionais a execução dos projetos. Por outro lado, o gerente de projeto tem poderes amplos, de forma que reporta diretamente aos executivos e comanda a execução dos trabalhos utilizando recursos “emprestados” pelos grupos funcionais pelo tempo necessário para o desenvolvimento de um produto. A principal função do gerente de projeto é planejar. Na prática, entretanto, atua também como mediador, certificando-se que, diante de conflitos entre os membros da equipe, os conceitos fundamentais estabelecidos para aquele projeto sejam priorizados (CLARK e FUJIMOTO, 1995).

A simples escolha de um gerente de projeto pode não significar que os problemas de controle e monitoramento estejam resolvidos definitivamente. Tal iniciativa deve vir acompanhada do estabelecimento de instrumentos que possibilitem que uma atividade possa ser avaliada quantitativamente, o que permite que a tomada de decisão fundamentada em valores quantitativos ocorra. A analogia que se faz é a de uma aeronave sem instrumentação. Não se pode tomar decisões com base em suposições ou estimativas subjetivas. Se o altímetro indica baixa altitude, o piloto deve ser capaz de realizar a leitura e tomar a decisão de efetuar a manobra para alcançar mais altitude. Da mesma forma os indicadores de projeto devem ser utilizados para fundamentar decisões da equipe de desenvolvimento (KAYDOS, 1999). Ao conjunto de indicadores de projeto utiliza-se a denominação **Sistema de Métricas**.

PATTERSON (1993) indica alguns atributos-chave para que um sistema de métricas não produza efeitos inesperados:

a) **relevância** – A métrica deve fornecer informação clara que enfoque os fatores importantes para a tarefa em questão. Alguns sistemas impõem métricas que necessitam de uma coleta exaustiva de dados, que podem ser apenas parcialmente relevantes. Um conjunto bem suportado e efetivo de métricas deveria, em cada caso, ser relevante para algum aspecto importante da função que está sendo medida.
b) **amplitude** – O conjunto de métricas deve tornar visíveis todos os fatores importantes com ênfase balanceada. Caso um seja deixado de

fora, o sistema eliminará o parâmetro omitido de forma que outros sejam melhorados. Por exemplo, se o *time-to-market* for o indicador mais importante, e se esqueça de incluir outro para qualidade do produto, é previsível o que acontecerá com a qualidade. Todos os aspectos importantes da operação sob análise deveriam ser medidos. O objetivo é um número mínimo de métricas, mas nada que seja crítico ao sucesso deveria ser omitido.

c) **tempo de resposta** – O tempo de resposta é função da rapidez com que o negócio possa mudar. As métricas em qualquer atividade de negócio devem ser suficientemente rápidas, a tal ponto de permitirem decisões relacionadas com a situação corrente da atividade em questão. As medidas não precisam refletir o estado corrente precisamente, mas deveriam propiciar, em tempo suficiente, a tomada de decisões corretas e eficazes.

d) **elegância** – Métricas podem ser uma carga extra à organização, gerando trabalho extra significativo, pois precisam de esforço para ser coletadas, reportadas, compiladas, monitoradas e arquivadas. O propósito de um conjunto de métricas é adquirir perspectiva sobre o desempenho real de um processo de negócio. Conhecimento é o objetivo, não os dados ou relatórios elaborados. Se a métrica for concebida com elegância, alcançará um nível máximo de perspectiva com uma quantidade mínima de dados.

É fundamental que se compreenda que o perfeito entendimento por parte dos envolvidos de que o desenvolvimento é em processo e que, como processo, pode e deve ser modelado (PATTERSON, 1993; GOLDENSE, 1997b). Esta idéia precede a utilização de métricas. Da mesma forma que as métricas, a modelagem do processo de desenvolvimento é essencial para a condução de um projeto. A partir de um modelo representativo do processo, é possível planejar, de forma que se garanta a qualidade da execução, ao mesmo tempo em que se estabelecem as estratégias de paralelismo entre

tarefas, respaldadas por um fluxo conhecido de informações. Uma vez capturado o fluxo de trabalho, pode-se partir para a sua reengenharia, otimizando os processos e mesmo implementando ferramentas que permitam a sua fácil verificação e monitoramento e que permitam que, a qualquer momento, uma avaliação de seu progresso. Em âmbito acadêmico, há diversas iniciativas de cooperação com a indústria no sentido de se modelar o processo de desenvolvimento de produtos de forma a otimizá-los, tais como as conduzidas por EPPINGER et al. (1994), EPPINGER, NUKALA e WHITNEY (1997), SMITH e EPPINGER (1997), SMITH e EPPINGER (1998), e SMITH e MORROW (1999). Por outro lado, EHRLENSPIEL, GIAPOULIS e GÜNTHER (1997), FRANKENBERGER e AUER (1997), e SMITH e TJANDRA (1998) realizaram trabalhos no sentido de modelar o comportamento de equipes de desenvolvimento de forma a extrair conclusões que permitam adequar os métodos de gerenciamento hoje existentes.

2.9 SISTEMAS EXISTENTES

Os estudos e protótipos de sistemas de informação que suportam ambientes de trabalho colaborativo estão presentes na literatura desde o início da década de noventa. Não há intenção na presente pesquisa de classificar todos os sistemas existentes, porém a Figura 5 traz um comparativo que objetiva estabelecer critérios para a classificação de sistemas, e por sua vez, justificar a necessidade pelo sistema aqui proposto.

Inicialmente, a Figura 5 classifica os sistemas de acordo com as funcionalidades que fornecem. Desta forma pode-se avaliar se um sistema proporciona um ambiente colaborativo, do tipo CSCW síncrono ou assíncrono, se adiciona funcionalidade de sistemas de *workflow* com ou sem captura de métricas, se permite o armazenamento de dados de projeto em sistemas do tipo PDM, e se permite o gerenciamento de recursos (humanos, tempo, ferramentas, entre outros). De uma forma geral, os melhores sistemas ora permitem uma ou outra funcionalidade. Outros

sistemas, cujas descrições não permitem que se obtenha detalhes, apenas propõem que todas as funcionalidades estejam presentes.

Em seguida, os sistemas pesquisados foram avaliados utilizando como base o trabalho de KING e MAJCHRZAK (1996), no qual vários deles foram analisados quanto aos fatores humanos necessários para o trabalho colaborativo. Dentro deste enfoque, pode-se questionar a presença de *wrappers*[8] conceituais, cuja função seria a de capturar as informações nas realidades contextuais em que a ferramenta de ES esteja sendo utilizada, e utilizá-las para modificar a ferramenta de acordo com as necessidades do usuário. Da mesma forma, KING e MAJCHRZAK sugerem que sejam concebidas ferramentas distintas para as fases conceitual e de detalhamento. Segundo os autores, as necessidades são tão distintas, em termos de cooperação, que justificam a existência de dois ambientes. Finalmente, há ainda de se diferenciar sistemas que permitam a captura de conhecimento, tomada de decisão e que proporcionem visões distintas aos participantes das equipes interdisciplinares, segundo os níveis de abstração (ver 2.8.1) em que trabalham. Percebe-se claramente que estes fatores estão ainda por serem aplicados, o que indica uma grave falha dos sistemas existentes nestes quesitos.

Um último critério de avaliação utilizado na Figura 5 é o da tecnologia aplicada em tais sistemas. De uma forma geral, se tratam de sistemas distribuídos e, como tais, a tendência é a adoção de plataformas e padrões comuns como XML, CORBA (CHEE e GOH, 1998; KIM e CHUNG, 1998) e J2EE. Como meio, são utilizadas as *intranets* corporativas e a própria *internet* para compartilhamento de informações e interações entre usuários. Os sistemas mais antigos não tinham a preocupação com a utilização de padrões amplamente disseminados, o que se mostrou essencial ao longo da última década, em função da utilização de diferentes sistemas operacionais, tais como HP-UX, Solaris, Windows e Linux, entre outros. Sistemas proprietários apresentam altos custos de atualização e suporte técnico e se apresentam

[8] *Wrappers*: elementos de síntese.

como uma ferramenta **a mais** a ser dominada pelo usuário.

Após uma análise do resultado da aplicação destes critérios, conforme tabulado na Figura 5, percebe-se que há uma oportunidade para desenvolvimento de um sistema capaz de fornecer uma solução melhor para o gerenciamento e prática da ES para o desenvolvimento de novos produtos. Há a necessidade de se inserir funcionalidades que permitam a captura de métricas. Por outro lado, os sistemas a serem desenvolvidos deverão oferecer pelo menos mecanismos de captura de conhecimento e suporte a diferentes níveis de abstração. Por último, com relação à tecnologia adotada, deve contemplar os mais recentes desdobramentos para sistemas em nível corporativo, levando-se em conta fatores como facilidade de integração, facilidade de operação e desempenho.

Figura 5 - Comparação entre sistemas existentes

			MANSFIELD *et al.* (Universidade de Queensland, Austrália 1999)	KIM e CHUNG (Instituto de Pesquisa em Engenharia de Sistemas, Coréia do Sul, 1998)	BAILEY e RUCKER (Universidade do Estado do Arizona, EUA, 1998)	MAHER e RUTHERFORD (Universidade da Austrália, Austrália, 1997)	CHEE e GOH (Universidade Tecnológica de Nanyang, Cingapura, 1998)	THRONICKE e SCHOLZ (Universidade de Paderborn, Alemanha, 1998)	ANUPAM e BAJAJ (Universidade Purdue, EUA, 1994)	DANGELMAIER *et al.* (Universidade de Paderborn, Alemanha, 1999)
Funcionalidades	Ambiente colaborativo	Síncrono	X	X	X	X		X	X	
		Assíncrono	X	X	X	X		X	X	
	Workflow	Sem métricas	X	X	X		X	X		X
		Com métricas			X					X
	Product Data Management		X	X	X	X	X	X	X	X
	Gerenciamento de recursos				X					
Fatores Humanos	*Wrappers* contextuais									
	Suporte a Projeto	Conceitual								
		Detalhado		X		X	X		X	
	Suporte a captura de conhecimento									
	Suporte a tomada de decisão									
	Suporte a níveis de abstração		X							
Tecnologia	Baseado em *Web*		X	X	X		X	X		X
	Com aplicativos comerciais				X	X	X	X		
	CORBA		X	X	X		X	X		
	JAVA *Scripts, Applets*		X	X	X		X	X		X
	Baseado em STEP			X			X			

3 METODOLOGIA

3.1 DELIMITAÇÃO DA PESQUISA

Esta pesquisa foi direcionada ao setor produtivo que desenvolve e comercializa bens de alto teor tecnológico, tais como o setor automotivo (automóveis, caminhões, tratores e outros), eletro-eletrônicos de consumo geral (áudio e vídeo, eletrodomésticos e outros), informática e telecomunicações (microcomputadores, centrais telefônicas, PABX e outros), bens de produção e medição (instrumentos de precisão, máquinas ferramenta e outros) e sistemas de grande porte (geradores, turbinas e outros). De maneira específica, o setor de telecomunicações foi focado para efeito de experimentação na pesquisa.

3.2 LIMITAÇÕES DO TRABALHO

É importante ressaltar que esta pesquisa não pretendeu desenvolver novas tecnologias de informática destinadas a sistemas de informação, e sim aplicar os recursos disponíveis na atualidade, bem como levou em consideração a situação real no âmbito das telecomunicações no Brasil. Desta forma, adotou-se, como base de desenvolvimento, componentes que, em função de seus custos, não tornassem a utilização do sistema proposto proibitiva a pequenas e médias empresas, tanto no aspecto de seu valor para aquisição no mercado, quando nos custos de implementação e suporte.

3.3 CLASSIFICAÇÃO DA PESQUISA

Ao objetivar a geração de conhecimento, através de técnicas e ferramentas, direcionados à solução do problema específico de adaptação da estrutura de trabalho para enfrentar as novas realidades de competição global que impõem reduções de

prazo, custo e melhoria de qualidade no desenvolvimento integrado de produtos de alto teor tecnológico, a presente pesquisa apresentou **natureza aplicada**. Segundo SILVA e MENEZES (2001, p. 20), esta envolve verdades e interesses locais. Neste caso, trataram-se dos interesses específicos das empresas envolvidas direta e indiretamente (clientes e fornecedores) no processo inovativo de desenvolvimento de novos produtos.

Com ênfase no processo de desenvolvimento de produto e seu significado, para o qual foram analisados dados indutivamente, esta pesquisa utilizou uma **abordagem qualitativa**. A coleta de dados foi feita a partir do ambiente natural onde o DIP ocorre, e neste caso o pesquisador foi o instrumento-chave (SILVA e MENEZES, 2001, p. 20), ao interpretar os fenômenos encontrados e atribuir-lhes significado. Não houve a utilização de métodos e técnicas estatísticas.

A presente pesquisa teve caráter **exploratório-descritivo** quanto aos seus **objetivos**, na medida em que envolveu levantamento bibliográfico, entrevistas com pessoas que tiveram experiências práticas com o problema pesquisado e análise de exemplos que estimularam a sua compreensão (cenários de DIP), além de ter lançado mão de técnicas padronizadas para coleta de dados durante a etapa de modelagem de processos e informação. Por outro lado, esta pesquisa envolveu o uso de questionários e observação sistemática aplicados a uma determinada população (equipe de desenvolvimento interdisciplinar) e fenômeno (DIP), com a finalidade de descrever as características do relacionamento entre os membros e estabelecer relações entre variáveis (atividades e elementos de informação, entre outras).

Do ponto de vista da adoção de **procedimentos técnicos**, a presente pesquisa apresentou características que a classificam como de tipo **bibliográfico**, de **levantamento**, **pesquisa-ação**, **participativa** e **experimental** (SILVA e MENEZES, 2001, p. 21 e 22).

O caráter bibliográfico deveu-se ao fato de ter-se procurado levantar na literatura todos os aspectos relevantes para uma implementação bem sucedida da sistemática da ES, evidenciados através das diferentes abordagens até hoje encontradas

na literatura. Através da observação dos diferentes enfoques utilizados até hoje, a pesquisa procurou sintetizar os aspectos mais relevantes de forma que pudessem ser materializados mediante a concepção de uma plataforma computacional de trabalho. Por ter sido elaborada a partir de material já publicado, constituído principalmente de livros e periódicos, esta pesquisa também apresentou características de levantamento.

Por outro lado, a presente pesquisa envolveu a interrogação direta das pessoas participantes do DIP, cujo comportamento se desejava conhecer. Por este motivo também apresentou características de pesquisa-ação. Não menos importante, este estudo também teve um caráter participativo, pois para a sua condução, necessitou do envolvimento direto do pesquisador no sentido de identificar, juntamente com profissionais de equipes de engenharia, as práticas correntes e aquelas desejáveis para a plena utilização dos preceitos da ES no desenvolvimento de produtos. Foi necessário que o pesquisador transmitisse de forma objetiva a conceituação do tema, a fim de justificar as práticas de ES já utilizadas e permitir que cada profissional colocasse suas dificuldades no exercício do desenvolvimento do produto, causadas muitas vezes pela não observância dos conceitos da ES. Os profissionais consultados foram expostos ao escopo do trabalho, de forma que compreendessem a amplitude e os propósitos deste esforço que, em síntese, sugere a adoção de uma organização do trabalho que facilite a prática da ES, culminando em melhorias dos parâmetros de avaliação de equipes de desenvolvimento sob a forma de menores prazos e custos e maior qualidade dos produtos concebidos.

Finalmente, esta investigação teve caráter experimental, uma vez que lançou mão de ferramentas de tecnologia da informação para apresentar um sistema computacional novo (uma nova variável) capaz de influenciar o trabalho colaborativo, sistematizado e coordenado para o desenvolvimento de produtos nas corporações (objeto de estudo), dentro do qual convergem as áreas de administração da produção e ciência da informação para a solução de um problema tipicamente social.

3.4 PROCEDIMENTO ADOTADO

A presente pesquisa foi conduzida em três etapas distintas, porém intrinsecamente associadas:

a) Levantamento de abordagens e melhores práticas;
b) Construção de modelos, subdividida em:
   - Modelo de processo;
   - Modelo de informação e;
   - Sistema de métricas;
c) Implementação do sistema, subdividida em:
   - Especificação;
   - Modelagem e;
   - Codificação.

Durante os estudos de revisão da literatura procurou-se identificar as diferentes abordagens do tema ES, a fim de considerá-las no ambiente computacional que foi concebido posteriormente. Desta forma, as recomendações levantadas em cada abordagem, referentes a o uso extensivo de **ferramentas computacionais** de auxílio a projetos (CAx), busca pela **integração** entre membros da equipe de desenvolvimento, utilização de **controle** de projetos mediante o uso de métricas e práticas de **gerenciamento** do processo de desenvolvimento foram consideradas para a especificação do sistema (ver 2.6 e 3.4.4).

3.4.1 Modelagem de processo

A etapa de modelagem de processo teve como propósito identificar a melhor forma de capturar um processo de desenvolvimento suficientemente maduro com relação às melhores práticas da ES, de forma que o mesmo pudesse ser utilizado através de uma ferramenta computacional. Dentro de um contexto específico (ver 3.1), o fluxo de trabalho e as interações que ocorrem num ambiente de ES foram documentadas. Cada modelo de processo em particular faz parte de um novo **cenário**,

segundo o qual projetos devem ser instanciados.

Desta forma, um cenário poderia ser o do desenvolvimento de foguetes para lançamento de satélites, outro poderia ser para o desenvolvimento de brinquedos feitos de peças plásticas, e assim por diante. É possível conceber novos cenários para que sejam incorporados à plataforma, desde que os modelos sejam elaborados, levando-se em conta aspectos como tipo de produto a ser desenvolvido, procedimentos em uso nas corporações, sistemática de trabalho e organização da empresa, procedimentos específicos ligados à garantia da qualidade, entre outros.

Para efeito de experimentação da sistemática adotada para levantamento do modelo de processos, foram capturados dois cenários: um referente ao desenvolvimento de armários para equipamentos de telecomunicações para exportação, e outro para produtos semelhantes, porém direcionados ao mercado doméstico brasileiro. Os processos capturados foram utilizados posteriormente para a implementação de funcionalidades do sistema que permitissem, tal como numa ferramenta de *workflow*, o acompanhamento das atividades, bem como o monitoramento das informações agregadas ao modelo de informação do produto ao longo do seu desenvolvimento.

Durante a etapa de modelagem de processos foram levantadas todas as transações que ocorrem entre os participantes da equipe de desenvolvimento, sejam eles internos ou externos à corporação. Para o levantamento dos procedimentos executados na empresa foram coletadas informações a partir de normas e manuais internos, sistemas informatizados de acompanhamento de trabalhos, instruções técnicas de trabalho e entrevistas com os membros mais atuantes das equipes de desenvolvimento.

Nas entrevistas realizadas, coube ao entrevistado descrever as atividades que realiza num projeto de desenvolvimento de produto a forma como executa seu trabalho, indicando as informações que compartilha, as obrigações que tem do ponto de vista de documentação, as dificuldades que percebe para levantar as informações que necessita e as ferramentas que dispõe para executar sua função. Por outro lado, o

entrevistador procurou direcionar as perguntas de forma a identificar as características mais importantes para a preparação dos modelos. O Apêndice 1 traz o modelo de questionário utilizado para direcionar as entrevistas.

A fim de facilitar os trabalhos de modelagem do processo foi feita uma subdivisão do modelo em níveis de abstração em relação aos quais as tarefas são realizadas. Desta forma, optou-se por adotar cinco diferentes níveis de abstração, o que reflete a realidade de estrutura matricial de trabalho normalmente apresentada em empresas de porte similar ao da empresa colaboradora focada, quais sejam:

- Nível 0: coordenador de projeto;
- Nível 1: gerentes de áreas internas de desenvolvimento;
- Nível 2: projetistas internos;
- Nível 3: líderes de equipes colaboradoras externas e;
- Nível 4: projetistas de equipes executoras externas.

As informações coletadas durante a etapa de modelagem de processo foram agrupadas por nível de abstração, dependendo da perspectiva e responsabilidades que os membros da equipe têm em relação ao produto. Os entrevistados foram escolhidos conforme suas responsabilidades nas equipes de projeto. Para os níveis de abstração 0 e 1, duas pessoas foram entrevistadas. Para os níveis de abstração 2, 3 e 4, pelo menos dois representantes de cada nível foram consultados, cada qual responsável por subsistemas distintos do produto. As entrevistas foram realizadas entre os dias 26 de janeiro e 7 de maio de 2001. As mesmas foram armazenadas em fitas de áudio para poderem ser consultadas a qualquer momento durante a elaboração dos modelos. Também houve consulta aos entrevistados por correio eletrônico, conforme a necessidade durante a etapa de modelagem.

Para a captura dos modelos de processo, propriamente dita, a ferramenta **ProSLCSE/Fun** (AMBIÊNCIA SISTEMAS DE INFORMAÇÃO LTDA., 2003), disponibilizada pelo Laboratório Olympus da PUC-PR foi inicialmente adotada, em virtude de estar também sendo utilizada para modelagem de processos em outros projetos de cooperação conduzidos pela empresa colaboradora. Foi utilizada de janeiro

a junho de 2001, paralelamente ao levantamento de informações por entrevistas. Nesta ferramenta, a modelagem de processo é feita através da sua representação em quatro eixos: tarefas, insumos necessários, entradas e saídas, e referências. As tarefas são representadas por elipses. Elipses sombreadas correspondem a tarefas que são decompostas posteriormente no fluxo em sub-tarefas e assim por diante. A figura de uma pessoa vinculada a uma tarefa corresponde à representação de recursos humanos necessários. As entradas e saídas de cada tarefa correspondem à alimentação e aos produtos gerados e são representados por cilindros. Os retângulos contendo linhas horizontais são representações das referências utilizadas para a realização da tarefa em questão, tais como normas e relatórios. Há também ícones que representam ferramentas necessárias à realização das tarefas, tais como computadores e *software*. A Figura 6 traz um exemplo de representação de processo através da ferramenta **ProSLCSE/Fun**.

Figura 6 - Modelagem na ferramenta ProSLCSE/Fun

Posteriormente, a partir de junho de 2001, optou-se pela utilização da ferramenta **ProSim** versão 6.0 (KNOWLEDGE BASE SYSTEMS, INC., 2000a), para

recapturar o modelo de processo dentro do método IDEF3 (MAYER et al., 1995). Tal redirecionamento ocorreu em função de:

- dificuldades para utilização da ferramenta **ProSLCSE/Fun**, que só poderia ser utilizada na PUC-PR;
- alto custo para obtenção de uma licença para utilização fora da PUC-PR e;
- ausência de facilidades para aproveitamento dos fluxos gerados pela ferramenta no sistema a ser desenvolvido.

Também foi percebido que o nível de detalhamento inicialmente adotado para o modelo criado na ferramenta **ProSLCSE/Fun** havia sido muito alto, o que traria um efeito de rejeição imediato dos usuários ao sistema de informação proposto. Desta forma, uma vez que a ferramenta utilizada na captura seria substituída, e os arquivos dos fluxos não poderiam ser aproveitados diretamente, optou-se também pela simplificação do modelo de processos, de forma que nem todas as tarefas anteriormente modeladas fossem representadas nos novos fluxos de trabalho. Além disso, a metodologia IDEF3, desenvolvida especificamente para documentar processos que utilizam a sistemática da ES, se mostrou mais adequada que o padrão proprietário da ferramenta **ProSLCSE/Fun**.

Posteriormente, foi utilizado um pacote de ferramentas opcional denominado **ProSim Option Pack** (KNOWLEDGE BASE SYSTEMS, INC., 2000b) que contém os módulos **ProjectLink** e **Process Knowledge Builder**. O módulo ProjectLink foi aplicado para converter diretamente de fluxos de trabalho levantados em arquivos compatíveis com o aplicativo de gestão de projetos Microsoft Project 2002 (MICROSOFT CORPORATION, 2002). Já o módulo Process Knlowledge Builder permitiu a geração de um dispositivo de navegação pelo fluxo de trabalho através de um *applet* Java.

O termo IDEF foi cunhado no programa da Força Aérea norte-americana denominado *Integrated Computer-Aided Manufacturing* (ICAM), que desenvolveu o primeiro método de definição ICAM, também denominado IDEF. Segundo MAYER et al. (1995), o desenvolvimento continuado da tecnologia IDEF apóia uma estratégia

global de fornecimento de métodos mutuamente suportados para a integração de empresas. Mais recentemente, com o foco expandido e o uso de métodos IDEF como parte de iniciativas de aplicação de ES, TQM e BPR, o acrônimo IDEF foi redefinido como uma família de métodos de definição de integração.

Ainda segundo MAYER et al., o método IDEF3 foi criado especificamente para capturar descrições de seqüências de atividades. O objetivo principal do IDEF3 é o de fornecer uma metodologia estruturada segundo a qual um especialista num determinado domínio pode expressar todo o conhecimento a respeito da operação de um sistema ou de uma organização em particular. A aquisição de conhecimento é possibilitada pela captura direta e natural de asserções sobre processos no mundo real e de eventos. Isto inclui a captura de asserções sobre objetos que participam do processo, objetos de suporte, e as relações de precedência e causalidade entre processos e eventos no ambiente.

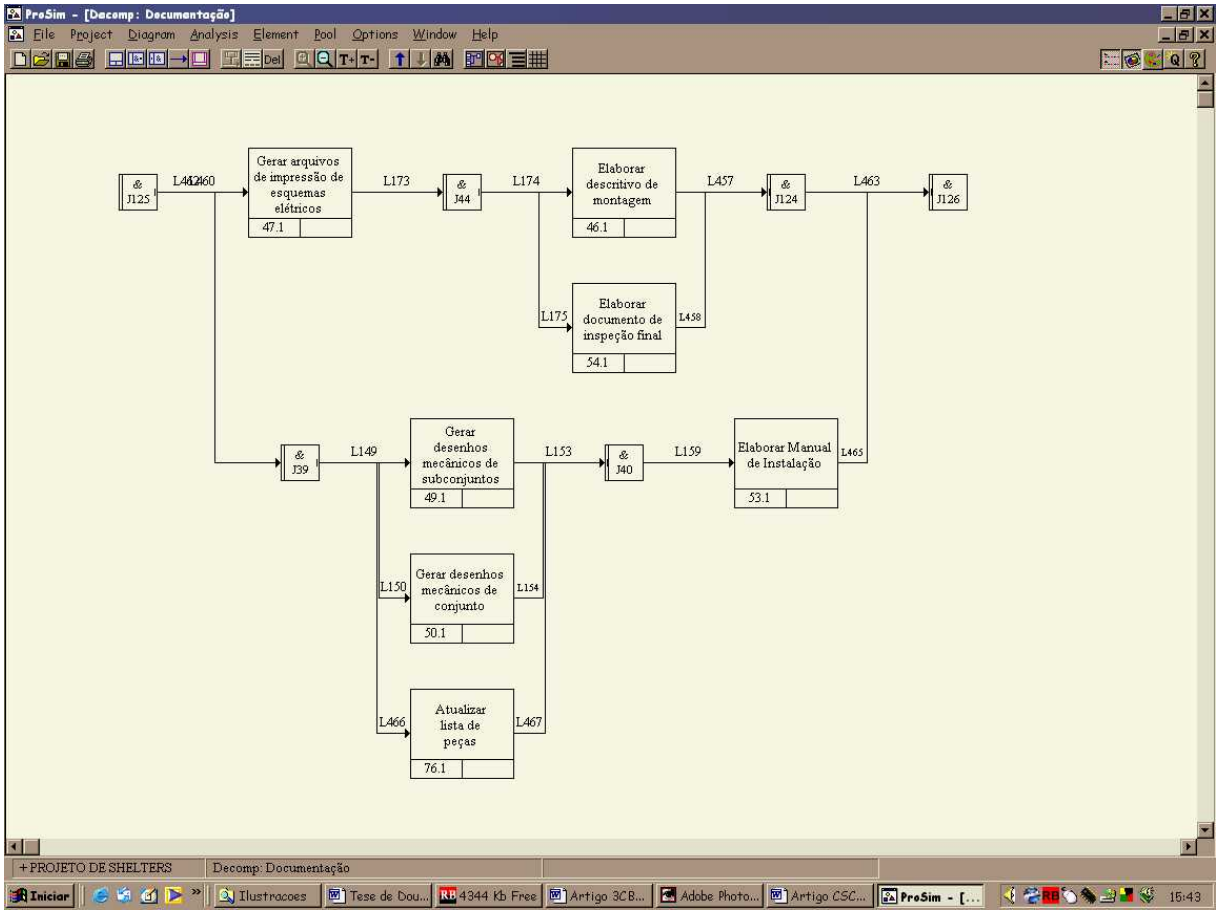
Na representação de processos utilizando o método IDEF3, as tarefas são modeladas em retângulos, que são conectados através de componentes lógicos do tipo “AND” e “OR”. As referências, tais como a infra-estrutura necessária, os operadores das tarefas e outras informações relevantes são registrados na ferramenta ProSim através dos conceitos de objetos, fatos e restrições. A Figura 7 apresenta um trecho da captura do processo de desenvolvimento de armários de telecomunicações seguindo a simbologia proposta pelo método IDEF3.

Numa etapa final da etapa de modelagem de processos, os fluxos de trabalho capturados através do método IDEF3 foram reapresentados aos colaboradores que haviam sido entrevistados no início do levantamento, para sua aprovação final. O modelo de processos para um dos cenários escolhidos pode ser visualizado na íntegra no Apêndice 3. A certificação dos modelos pelos entrevistados ocorreu em agosto de 2001, já em padrão IDEF3.

3.4.2 Modelagem de informação

Para o levantamento do modelo de informação, foram utilizados os mesmos dados através de entrevistas junto aos membros das equipes de desenvolvimento, assim como documentação de uso interno e sistemas de informação em uso (ver 3.4.1). O trabalho de captura do modelo transcorreu de fevereiro a maio de 2001.

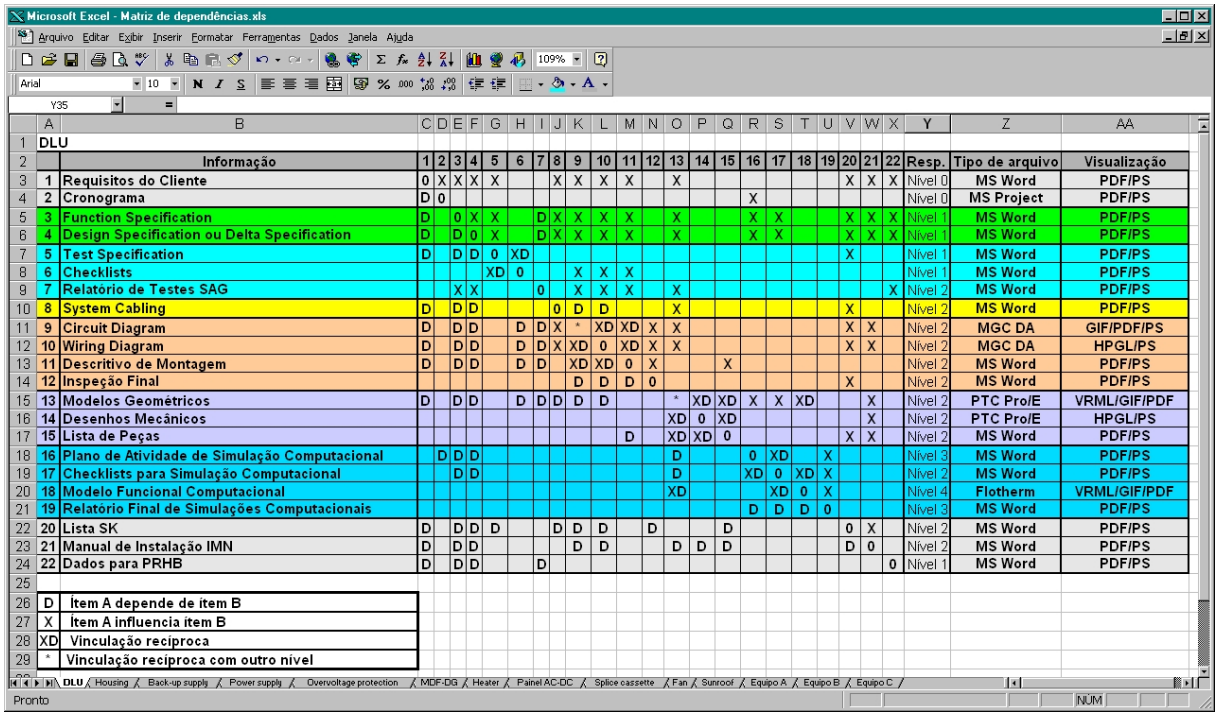
Figura 7 - Modelagem de processos através da ferramenta ProSim



Considerando que o modelo de informação definido para este trabalho representa de forma cronológica o que acontece com as informações que são agregadas ao produto, optou-se por representá-lo através de duas formas: uma matriz de dependências e um fluxograma de mudança de status da informação (OSTN). Posteriormente, estas formas de representação do modelo de informação foram convertidas no sistema computacional através de vinculações entre elementos de

informação, atribuições de status para descrição da situação do conjunto de informações e definição de níveis de acesso para o sistema de gerenciamento de dados de projeto, sistema este baseado em um banco de dados relacional.

Figura 8 - Matriz de dependências

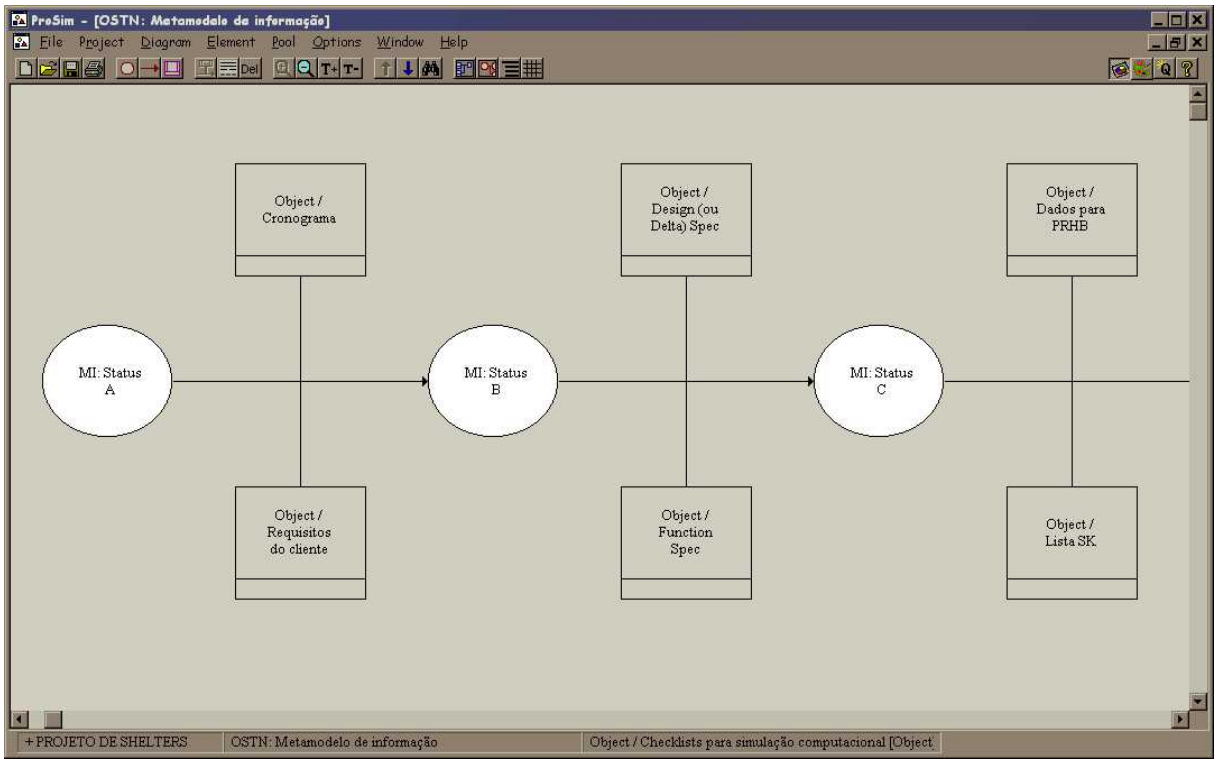


Similarmente ao modelo de processos, o modelo de informação faz parte de um cenário específico, que é utilizado para instanciar projetos semelhantes. Diferentes cenários possuem modelos de informação e processos distintos uns dos outros.

A matriz de dependências, cuja concepção é apresentada na Figura 8, é constituída de uma lista de arquivos ou documentos utilizados para a gestão, execução, intercâmbio de informações e armazenagem de dados. Os itens desta lista são referenciados em relação aos demais no sentido de indicar a sua situação de dependência e/ou influência. Trata-se de uma matriz tridimensional, uma vez que cada subsistema constituinte da árvore de decomposição do produto é representado através

de uma camada[9] e os elementos de diferentes camadas podem ser referenciados uns aos outros. A vinculação entre os elementos pode ser lida de forma que um elemento **dependa de outro** ou **influencie outro**. Assim, a matriz resultante é simétrica. A matriz de dependências foi baseada no conceito de Matriz de Estrutura de Projeto (DSM), apresentado por EPPINGER et al. (1994). As diferentes camadas de uma matriz de dependências podem ser visualizadas no Apêndice 7.

Figura 9 - Fluxograma de mudança de estado da informação



O fluxograma de mudança de status da informação (OSTN) para o cenário de desenvolvimento de armários de telecomunicações, apresentado na Figura 9, foi concebido utilizando-se a ferramenta **ProSim**. Consiste num diagrama que representa a "linha de montagem da informação", tal como descrito por PATTERSON (1993). O modelo de informação assume determinados estados (ou status), assim que os componentes especificados *a priori* são agregados ao longo do tempo.

[9] Representada por uma planilha distinta numa pasta de trabalho no *software* **Microsoft Excel**.

No cenário utilizado como piloto para concepção do sistema, foi necessário criar status intermediários que completassem uma seqüência lógica do começo ao final do ciclo de desenvolvimento, uma vez que nem todos os estados principais do modelo eram reconhecidos como tais pelos colaboradores entrevistados, ou mesmo através da documentação consultada.

Figura 10 - Associação entre tarefas e elementos de informação no *software* ProSim

Legend:
E = Entity
L = Location
Q = Queue
S = Resource
T = Transport
l = Logical
? = Conflict
- = Default

Processes	Objects	Aprovação de documentação para fabricação	Aprovação do produto	Arquivamento de dados	Catálogos de fornecedores	Checklists em Test Spec	Checklists para simulação computacional	Circuit Diagram do Back-up Supply	Circuit Diagram do DLU shelter (conjunto)	Circuit Diagram do Heater
	E	E	E	E	S	E	E	E	E	E
Documentar	L									
Especificar	L									
Homologar materiais e fornecedores	L									
Implementar	L								X	
Liberar	L			X						
Testar	L					X				

Além do status da informação, muitas empresas adotam o sistema de *gates* (ou marcos, ou ainda *milestones*) de aprovação de etapas intermediárias de projeto (ver 2.4). Similarmente ao OSTN referente a status de informações, pode-se criar outro relativo aos portais. O Apêndice 4 traz a modelagem de um OSTN de marcos e o Apêndice 5 apresenta o OSTN de status para o mesmo cenário.

Da mesma forma que no modelo de processos, a matriz de dependências e o fluxograma de estado da informação foram reapresentados aos colaboradores que

haviam sido entrevistados no início do levantamento, para sua aprovação final.

A união dos modelos de informação e processo ocorreu através da associação de elementos de informação às tarefas de um fluxo de trabalho. Desta forma, um colaborador que executa uma determinada tarefa, na verdade **trabalha** determinados elementos de informação, desde uma situação na qual nada existe até outra na qual a informação esteja suficientemente precisa.

O levantamento das associações entre tarefas e elementos de informação foi efetuado com base nas informações práticas fornecidas pelos projetistas, engenheiros e demais membros da equipe de desenvolvimento durante a realização das entrevistas. Para o trabalho de captura, existem recursos específicos na ferramenta ProSim. Um segmento do trabalho de vinculação entre tarefas e elementos pode ser visualizado na Figura 10 e, de forma mais abrangente no Apêndice 6.

3.4.3 Sistema de métricas

Um sistema de métricas bem especificado permite o monitoramento das atividades da equipe de desenvolvimento, permitindo que indicadores sejam levados em consideração para a correta tomada de decisões. Este monitoramento deve apresentar confiabilidade, relevância e tempo de resposta compatível com ações corretivas. Para tanto, é importante que os indicadores sejam levantados a partir das ações dos participantes da equipe ao executarem tarefas sobre a plataforma computacional de trabalho.

O sistema de métricas escolhido deve ser confiável no sentido de informar os coordenadores de projeto quanto à real situação de utilização de recursos humanos e instrumentais, tempo e custo de desenvolvimento, assim como respaldar os membros da equipe de desenvolvimento para a tomada de decisões. Para tanto, é necessário que atributos de relevância, amplitude e tempo de resposta sejam levados em consideração (ver 2.8.3).

De acordo com o estudo realizado por LOCH, STEIN e TERWIESCH

(1996), existem 28 variáveis que medem o desempenho de processos de desenvolvimento. LOCH e seus colegas agrupam estas 28 variáveis em 4 categorias: foco e estrutura de P&D, gerenciamento de projetos, integração interdisciplinar, e aprendizado e gerenciamento de pessoas. Cada um destes grupos apresenta, segundo o estudo, correlação com indicadores de resultado dos setores de desenvolvimento, tais como liderança de mercado, taxa de inovação, novidade em linhas de produtos e projeto direcionado ao custo.

Na presente pesquisa, houve interesse em se limitar o número destas variáveis a fim de se tomar somente aquelas que mais se relacionam com os efeitos da aplicação da sistemática da ES sobre o resultado dos setores de desenvolvimento, que por sua vez afetam o desempenho geral das empresas.

A avaliação quanto à associação entre as variáveis propostas por LOCH e seus colegas e os preceitos da ES foi feita com base na literatura, e não através de estudos estatísticos, conforme a metodologia utilizada pelo autor naquele estudo. Naturalmente as descobertas específicas do autor para a indústria de computadores e telecomunicações são levadas em consideração nesta investigação, por se tratar da sub-área escolhida para experimentação.

As variáveis que compõem o sistema de métricas implementado com base nas operações dos participantes da equipe de desenvolvimento sobre a plataforma de suporte são:

- Índice de desvio entre cronograma planejado e executado (IDC);
- Índice de paralelismo entre fases de projeto (IPF);
- Número de alterações de especificação (NAE);
- Número de sincronizações de bases de dados de projeto (NSD);
- Índice de integração com a manufatura (IIM) e;
- Índice de integração com fornecedores (IIF).

Pelos critérios estabelecidos por LOCH et al., as métricas IPF, NAE e IIF têm forte correlação com a taxa de inovação, enquanto as métricas IDC, NSD e IIM têm correlação com a liderança de mercado.

3.4.4 Especificação do sistema

Antes da etapa de modelagem de informação e processos, havia a idéia de se utilizar facilidades de customização e adaptação de ferramentas (PDM, CSCW, gestão de projetos, *workflow* e assim por diante) disponíveis no mercado. Com o intuito de integrá-las e compor então um sistema maior, as características particulares de cada componente deveriam ser compreendidas, assimiladas e manipuladas. Por um lado, esta estratégia não exigiria um esforço de programação muito grande, porém, por outro lado necessitaria da exploração de todos os mecanismos de customização de tais ferramentas.

Esta estratégia se mostrou inviável, pois os fabricantes não forneceriam os componentes dentro das expectativas de custo para a pesquisa, ou mesmo não concordavam com tal nível de adaptação de sua funcionalidade. Muitas ferramentas de PDM executam funções de sistemas CSCW. Por sua vez, há sistemas CSCW que englobam operações de *workflow*. Há ainda ferramentas de gestão de projetos que incorporam funcionalidades de ferramentas do tipo *workflow* e CSCW, mas não permitem sua integração com ferramentas do tipo PDM. Verificações sobre os componentes inicialmente considerados para integração, conduzidas de outubro de 2000 a junho de 2001 indicaram que a manipulação necessária sobre os mesmos, a fim de executarem as funcionalidades desejadas, se mostrava complexa demais e impraticável dentro das restrições de custo e tempo existentes nesta pesquisa.

Diante deste impasse, optou-se pela especificação, modelagem e implementação de código próprio, utilizando os recursos de programação ao alcance. Tais recursos incluíam banco de dados relacional de domínio público, ferramentas IDE com licença de uso disponível para pesquisa e componentes de *software* de uso autorizado para demonstração ou comprovação de viabilidade técnica (*applets* Java).

O sistema de informação proposto foi então concebido a partir dos requisitos descritos a seguir, baseados nas tendências evidenciadas na literatura (ver 2.9).

(a) Arquitetura

Um sistema cliente-servidor deve integrar todos os colaboradores, dentro e fora da empresa anfitriã[10], que participem do ciclo completo de desenvolvimento de um produto (incluindo parceiros, fornecedores, representantes de áreas como *marketing*, fabricação, vendas, assistência técnica, entre outros). Cada integrante da equipe deve possuir direitos de acesso e manipulação de informações de projeto compatíveis com seu nível de abstração[11].

Ao mesmo tempo em que esta arquitetura permite a integração, o desempenho da mesma não pode ser penalizado pela centralização de informações. Em função do contexto atual, no qual várias funções importantes são executadas por indivíduos ou equipes dispersas, os arquivos grandes e de interesse local (que são atualizados com freqüência) não podem ser armazenados em locais remotos que não sejam atendidos por *links* com características adequadas de velocidade.

(b) Escalabilidade

O sistema deve ser capaz de armazenar simultaneamente informações referentes a diversos projetos em andamento. O volume total de dados a ser gerenciado pode chegar à ordem de *gigabytes*, sendo que alguns elementos do modelo de informação podem ser da ordem de centenas de *megabytes*.

(c) Organização de dados

O modelo de informação deve ser composto por elementos originados a partir dos mais diversos aplicativos, desde ferramentas CAD e CAE até simples arquivos de texto contendo especificações, relatórios de testes, entre outros. Os

[10] Empresa que conduz os trabalhos de desenvolvimento do produto e que o introduz no mercado.

[11] Ponto de vista e responsabilidades em relação ao produto.

elementos do modelo sofrem atualizações constantes ao longo do ciclo de desenvolvimento, à medida que novas informações são agregadas aos mesmos.

A organização dos dados do produto na base de dados deve refletir a decomposição do mesmo em subsistemas, da mesma forma como um automóvel é subdividido em suspensão, trem de força, carroceria, e assim por diante. Assim, os repositórios definidos para armazenar as unidades de informação (arquivos) devem ser agrupados por ramo da árvore de decomposição do produto.

(d) Segurança

Todas as informações de projeto devem ser protegidas contra acessos de pessoas não pertencentes à equipe de desenvolvimento. Em função da possibilidade da equipe estar geograficamente dispersa, se os dados trafegarem pela rede pública via *internet*, os mesmos devem ser protegidos através de mecanismos de criptografia.

(e) Interface

A interface deve ser baseada em navegadores comuns não proprietários a partir de qualquer localidade conectada à rede. *Plug-ins* específicos podem ser agregados aos navegadores, principalmente para permitir a visualização de documentos. Aplicativos podem ser abertos a partir da definição de MIME que caracteriza o tipo de arquivo relativo a cada item armazenado.

(f) Cópias de segurança

O sistema deve permitir facilmente a obtenção de cópias de segurança automáticas e sua recuperação conforme a necessidade, com a intervenção de administrador.

(g) Infra-estrutura

Parte-se do pressuposto que os *links* de acesso ao sistema são em banda larga, permitindo conexões com taxa de transferência mínima de 128 kbps (*eg*. ISDN, DSL, BWA e *cable modem*).

(h) Arquivos para visualização e arquivos fonte

Em função da heterogeneidade de aplicativos para manipular os elementos do modelo de informação, cada um deles deve ser associado fortemente a um item **anexo** para visualização, que deve ser de formato HTML, PDF, PS, VRML, HPGL ou outros que apresentem características similares. Para muitos dos integrantes da equipe basta a possibilidade de visualização e *markup* para participarem do trabalho de forma mais efetiva.

(i) Características de acesso

Cada usuário deverá acessar o sistema através de um **portal**. Neste, as informações de cada projeto no qual o usuário estiver envolvido ficam em evidência. Outras funcionalidades como notícias de projeto (*news*), acesso a referências, contatos, informações de projetos já encerrados, entre outras, podem ser disponibilizadas também.

(j) Histórico do trabalho

O histórico do trabalho deverá ser registrado em duas frentes: histórico das atividades ou tarefas e histórico das informações. O histórico das atividades deve armazenar o progresso das atividades e datas relevantes, como início, término e data de levantamento dos registros. Com base nestas informações é possível compreender como ocorrem desvios entre as atividades previstas e aquelas executadas. Portanto, cada tarefa poderá estar associada a outros registros de histórico, cada qual contendo

informações pontuais sobre a sua realização no tempo.

Por outro lado, o histórico das informações indica como o modelo de informação evolui no tempo. Este pode ser considerado de forma análoga a um automóvel na linha de montagem. Em cada nova etapa, componentes são adicionados de forma que no final o automóvel esteja completo e funcionando. No caso da informação, esta deve ser suficientemente completa para que o produto seja fabricado, mantido e tecnicamente assistido no pós-venda. Portanto os componentes do automóvel (elementos do modelo de informação) devem "encaixar" perfeitamente quando são montados (consistência dos elementos).

O mecanismo que garante a consistência dos elementos do modelo de informação deve ser equivalente a um mecanismo de versionamento. Cada *release* do projeto deve conter elementos de informação consistentes entre si, mesmo que individualmente estejam em versões diferentes. Tanto versões de elementos quanto *releases* de projeto anteriores devem ser mantidos na base de dados, pois podem ser relevantes caso se necessite resgatar o histórico dos mesmos.

Não obstante, o sistema deve ser concebido de forma a capturar o conhecimento agregado ao produto, mesmo que não esteja explícito nos dados. Tal conhecimento é igualmente importante, não só durante o desenvolvimento do produto, mas também para possibilitar o resgate posterior do histórico do mesmo. Assim, deve haver a possibilidade de registrar sessões de trabalho (*on-line*) ou reuniões (*off-line*) e associá-las aos modelos de processo (tarefas) e informação (elementos) utilizados.

(k) Gerenciamento de recursos

Deve ser possível acompanhar o andamento do projeto não só do ponto de vista de execução de tarefas no tempo e seu impacto na evolução dos dados, mas também em relação aos recursos previstos. Assim, é desejável que as informações referentes a cada tarefa sejam registradas diretamente da base de dados, assim como consultadas a partir da mesma. As responsabilidades sobre cada tarefa devem estar

consistentes com os direitos de acesso aos elementos manipulados ao longo daquela atividade, e parâmetros ajustáveis de custo e dedicação ao projeto devem permear os diversos componentes do sistema.

(l) Gerenciamento de dados de projeto

Os dados mestres devem ser mantidos com unicidade num repositório seguro, onde a sua integridade pode ser assegurada e todas as alterações sobre eles podem ser monitoradas, controladas e registradas. Entretanto, a consulta aos dados mestres pode ser liberada a usuários em diferentes níveis de abstração. Aos responsáveis pela execução de tarefas deve ser concedida a permissão para atualização dos dados referentes ao progresso das mesmas.

Alguns elementos do modelo de informação possuem forte vinculação entre si, de forma que uma atualização de um elemento deve provocar necessariamente a revisão de outro, para garantir a consistência da informação. Outros elementos podem ter vinculações mais fracas, de forma que ao usuário responsável pela atualização daquele elemento influenciado seja solicitada a verificação de consistência.

Manipulações de elementos de informação devem disparar uma série de ações para bloqueio ou desbloqueio de elementos vinculados, através de notificações aos responsáveis.

(m) Trabalho colaborativo

Algumas funcionalidades de sistemas do tipo *groupware* são desejáveis, tais como:

a) abertura de discussões sobre assuntos específicos do projeto que envolvam uma ou mais áreas de conhecimento, que sejam organizadas (por exemplo vinculadas a elementos do modelo) e que sejam passíveis de registro;

b) abertura de sessões de trabalho colaborativo sobre *whiteboards*[12], onde seja possível apresentar imagens, efetuar marcações, ou mesmo apresentar *displays* passivos de aplicativos que estejam sendo executados remotamente;
c) notificação resumida automática de todas as modificações/inclusões efetuadas na base de dados desde o último *logon* e;
d) marcação de elementos de forma que seja possível indicar a ciência do usuário sobre o conteúdo dos mesmos.

(n) Gerenciamento do fluxo de trabalho

O sistema deve permitir que os trabalhos sejam acompanhados por todos, conforme o modelo de processos levantado para o ciclo de desenvolvimento. Deve haver a possibilidade de visualização imediata da situação do processo em relação ao fluxo a ser seguido, utilizando para isso de um mecanismo de sinalização (cores ou símbolos).

Cada projeto deve ter um fluxo único que obedece ao modelo de processo especificado no cenário definido quando da criação do projeto. A visualização em forma gráfica do processo que está em progresso permite que facilmente os participantes da equipe se situem em relação ao todo do ciclo, podendo observar o que já foi feito e o que está por vir. Desta forma tomam conhecimento de que dados deverão estar prontos em quais etapas, ou se já deveria estar disponibilizada alguma informação preliminar para algum colaborador.

Atualizações sobre o progresso de tarefas devem disparar uma série de ações para bloqueio ou desbloqueio de tarefas vinculadas (paralelas ou subseqüentes), atualização de status ou mesmo ultrapassagem de marcos de projeto, conhecidos como *milestones*, através de notificações aos responsáveis.

[12] *Whiteboard*: painel eletrônico que aceita marcações indicativas de comentários.

(o) Captura de métricas

Métricas relativas ao desempenho da equipe por conta de sua interação e realização das metas previstas devem ser acessíveis a qualquer momento pelos usuários cujas atribuições necessitem destas informações, em forma gráfica, para que ações possam ser tomadas ainda enquanto o projeto esteja em andamento. As métricas devem ser capturadas diretamente a partir da base de dados e das operações executadas sobre a mesma.

3.4.5 Modelagem do sistema

Uma vez levantados os requisitos do sistema, ocorreu a modelagem em nível lógico do banco de dados relacional a ser implementado. Esta etapa transcorreu de setembro de 2001 a março de 2002. Foi de fundamental importância porque influiu diretamente na qualidade do sistema, na medida em que apresentou os relacionamentos e dependências entre cada componente de informação a ser trabalhado durante o desenvolvimento de um produto.

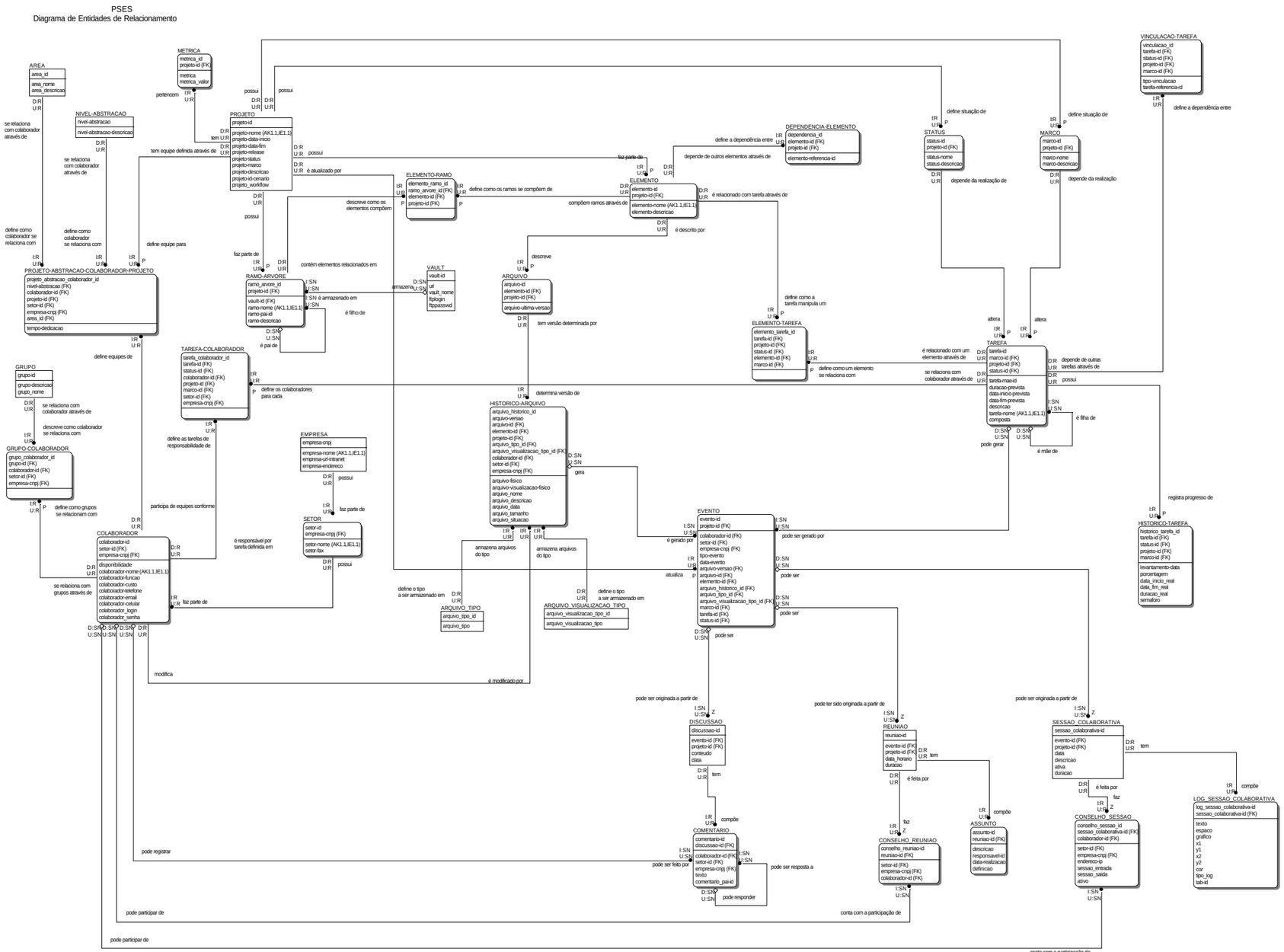
Para esta pesquisa, lançou-se a mão da ferramenta **ERwin** (LOGIC WORKS, 1997), que permitiu a construção do diagrama de relacionamento de entidades (DER), conforme apresentado na Figura 11. A própria ferramenta ERwin permite o aproveitamento quase que direto do DER para construir a base de dados através de funções que elaboram *scripts* SQL de forma automática. Para tanto é necessário que seja escolhido um banco de dados para o qual o modelo será direcionado.

Optou-se pelo banco de dados relacional PostgreSQL (DBEXPERTS, 2001), de domínio público e que cumpre todos os requisitos necessários para esta plataforma. Como esta alternativa não existe na versão 3.5.2 da ferramenta ERwin, foi gerado um *script* de criação do banco de dados em linguagem padrão SQL com pequenos ajustes[13], cuja transcrição se encontra no Apêndice 2. Quando o *script* é gerado, cada

[13] Substituição de **`go`** por **`;`**, de **`datetime`** por **`date`** e de **`binary()`** por **`bytea`**.

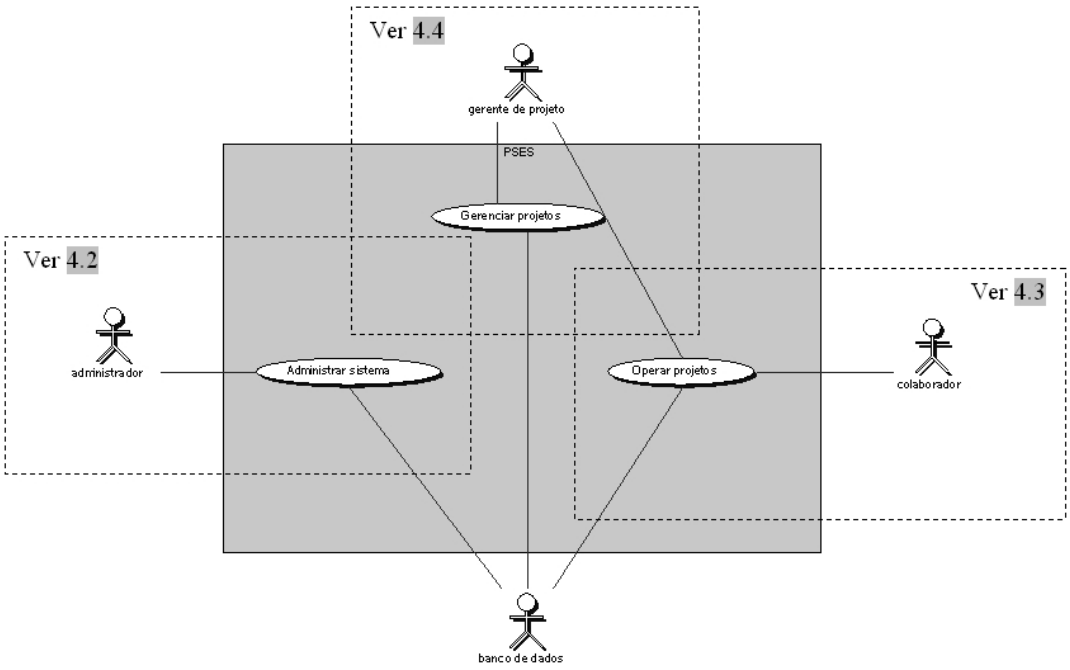
entidade do DER é convertida em uma tabela que, por sua vez, contém chaves primárias, secundárias e relacionamentos com outras tabelas conforme descrito no modelo. Posteriormente, verificou-se a necessidade de quebra de relacionamentos na tabela `evento`, procedimento este que foi adicionado manualmente conforme descrito nas últimas linhas do *script*.

Figura 11 - Diagrama de relacionamento de entidades (nível lógico)



A modelagem do DER permitiu que uma série de questões sobre a funcionalidade do sistema fossem discernidas. Este processo de análise não se encerrou ao final da criação da última entidade do DER, pois foi necessário promover várias alterações inclusive durante a etapa de codificação do sistema. Dentre as alterações mais significativas, pode-se destacar a modificação da tabela `historico_tarefa`, cujas colunas `data_inicio_real` e `data_fim_real` foram adicionadas posteriormente, durante a codificação.

Figura 12 - Modelo geral de caso de uso

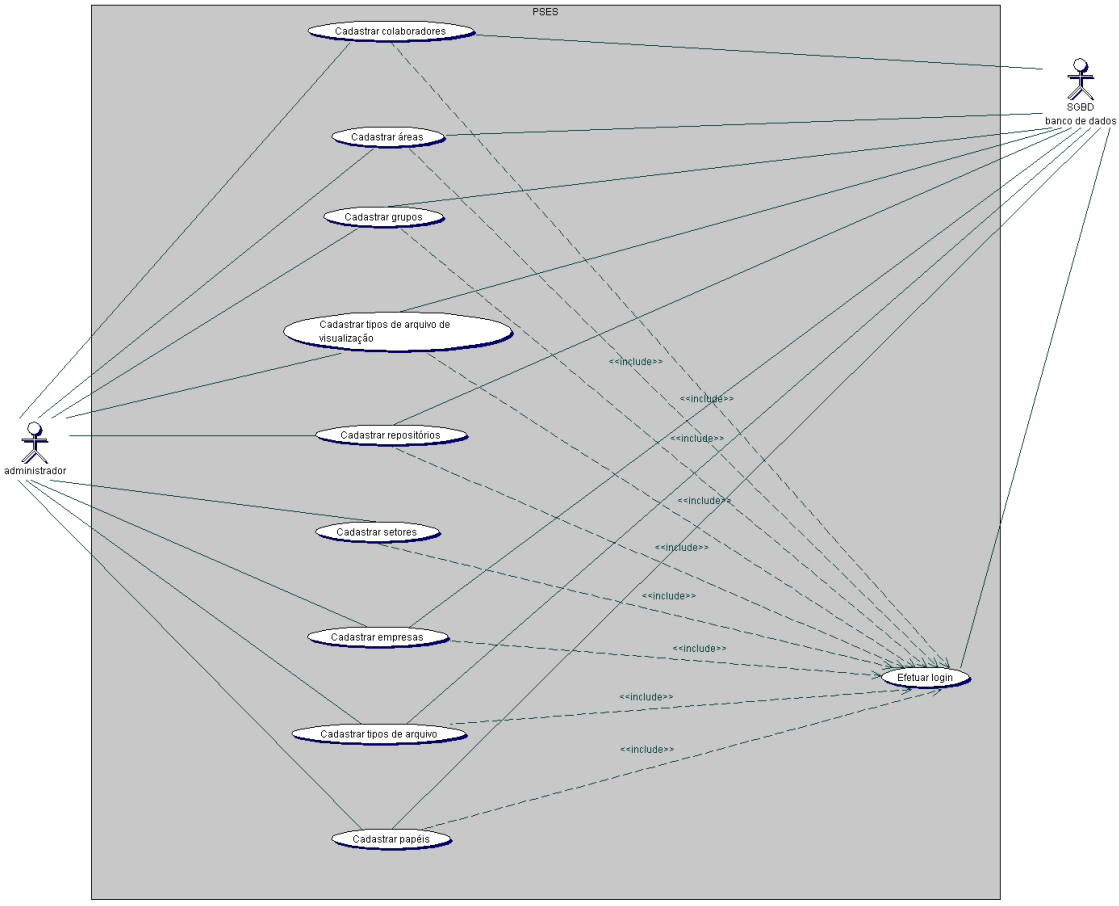


A etapa seguinte foi a de modelagem do sistema. Uma vez definido o paradigma de orientação a objetos (OO), para satisfazer muitos dos requisitos levantados para o sistema, optou-se pela modelagem da funcionalidade do mesmo através de UML (STADSZISZ, 2002), que é o padrão *de facto* utilizado pela indústria de *software* para especificar, visualizar, construir e documentar sistemas.

A modelagem do sistema foi feita inicialmente através de **casos de uso**. Os

modelos de casos de uso envolvem a determinação dos usos que o sistema terá, ou seja, do quê ele deverá fornecer como serviços. Desta forma, foi possível detectar três grandes modos de utilização: administração, gestão e operação. Estes modos são operados respectivamente pelo administrador do sistema de informação (doravante referenciado apenas como **administrador**), gerentes de projetos de desenvolvimento de produtos (doravante referenciados apenas como **gerentes**) e colaboradores da equipe de DIP. A Figura 12 traz uma ilustração do modelo em seu nível de topo. Para esta etapa da modelagem, foi utilizada a ferramenta **Together**, versão 6.0 (BORLAND SOFTWARE CORPORATION, 2002b). A notação segue o padrão UML.

Figura 13 - Modelo de caso de uso para administração



Na seqüência, foram refinados os casos de uso de administração (Figura 13),

gestão (Figura 14) e operação (Figura 15). Posteriormente, durante a etapa codificação, estes módulos foram interpretados como pacotes de um aplicativo *Web*[14].

Dentro do caso de uso de **administração**, o administrador do sistema pode intervir para cadastrar (incluir ou eliminar) empresas, setores, usuários, áreas, grupos, papéis, tipos de arquivo, tipos de arquivo de visualização e repositórios. Novos cenários, uma vez desenvolvido algum mecanismo para a sua inclusão, deverão ser igualmente adicionados pelo administrador.

O caso de uso de **gestão** prevê quatro utilizações básicas: configuração, avaliação, condução e acompanhamento de projetos. A configuração implica na criação do projeto com base num cenário pré-existente e definição da equipe de trabalho, responsabilidades, repositórios de arquivos e datas para realização de atividades. Estas informações poderão eventualmente ser modificadas, conforme a necessidade. A avaliação do projeto consiste na utilização dos recursos de métricas para monitoramento do mesmo, utilizando como critérios indicadores que demonstrem de que forma os preceitos da ES estão sendo seguidos. A condução do projeto permite que o gerente sincronize a base de dados através das atualizações de *release* e cronograma mestre. Atualizações de status e marcos de projeto são feitas automaticamente pelo sistema através da verificação das informações de progresso de tarefas inseridas pelos colaboradores individualmente. Finalmente, o modo de acompanhamento se dá através da verificação dos eventos do projeto e da sinalização da situação das tarefas no *workflow.*

O caso de uso de **operação** é realizado pelo colaborador membro de uma equipe de desenvolvimento. O colaborador pode acompanhar os eventos do projeto, verificar o progresso das tarefas, atualizar o progresso das tarefas sobre as quais tem responsabilidade, inserir a atualizar arquivos dos elementos de informação, abrir

[14] Consiste num aplicativo de arquitetura cliente/servidor que necessita de um navegador no lado do cliente e de um servidor *web* no lado do servidor. Aplicativos *web* podem ser disponibilizados a qualquer indivíduo que tenha acesso à *Internet*.

sessões de trabalho colaborativo, registrar e consultar reuniões, entre outras ações.

Figura 14 - Modelo de caso de uso para gestão

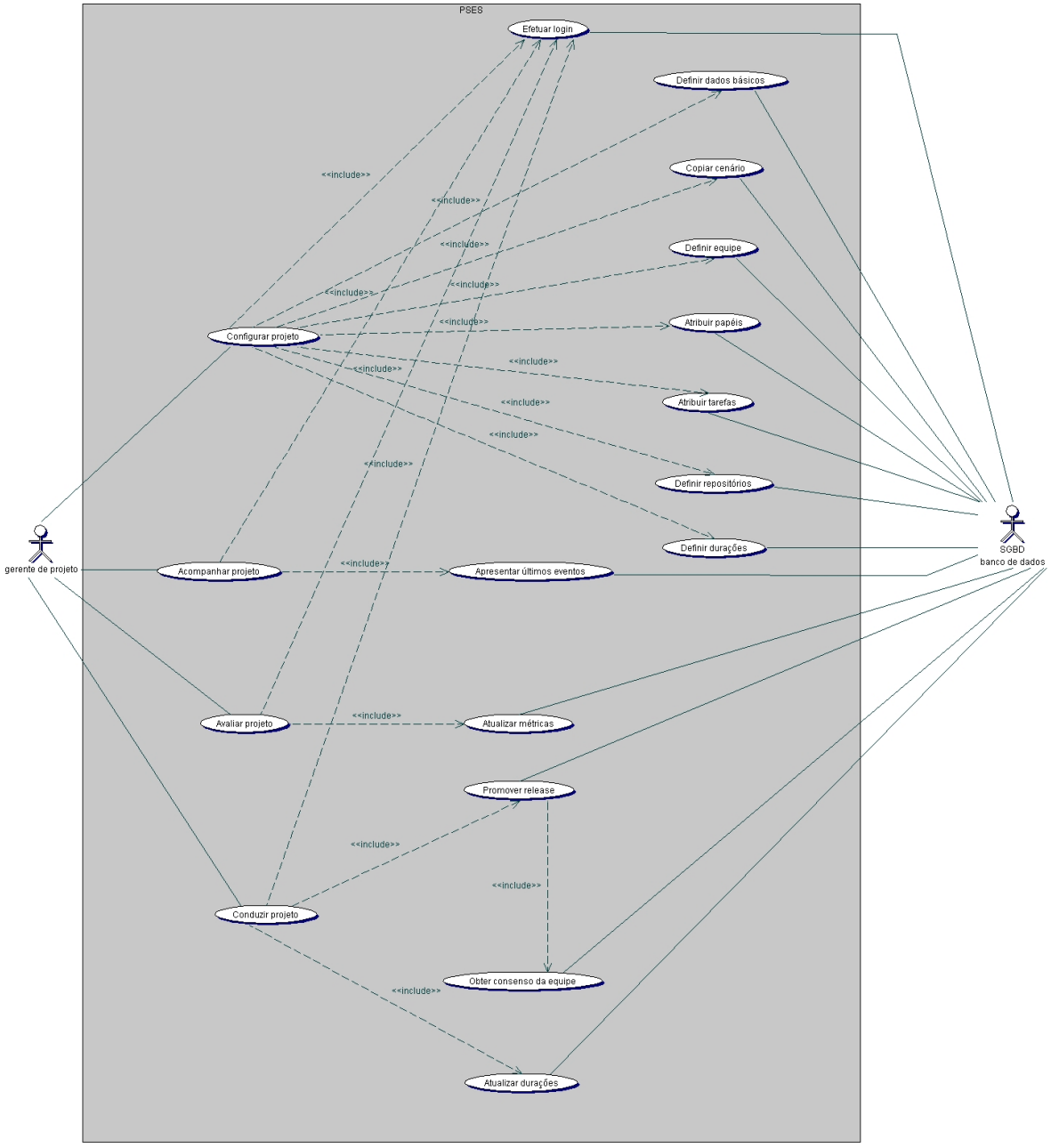
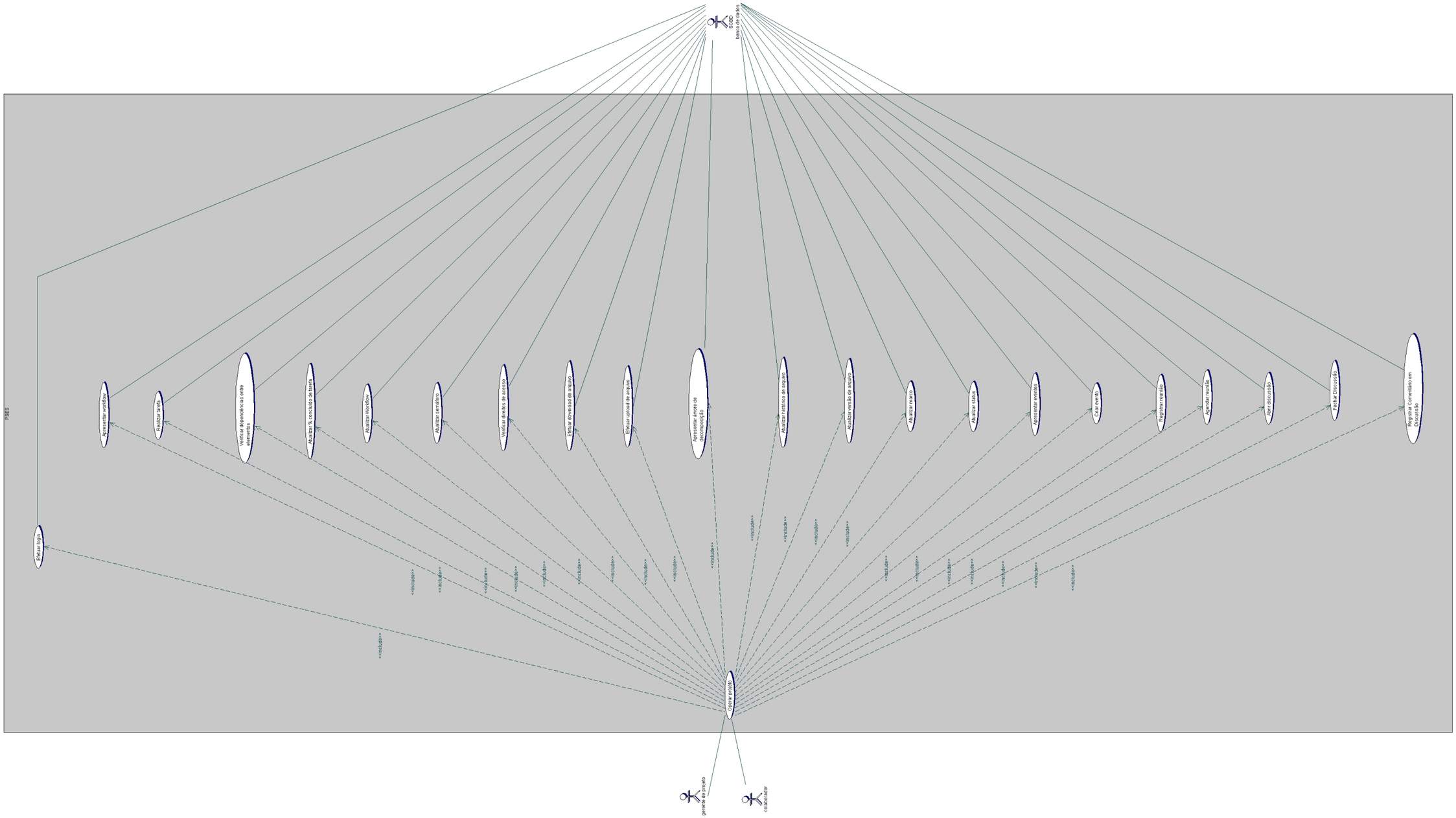


Figura 15 - Modelo de caso de uso para operação



Uma vez modelados os casos de uso, partiu-se para o levantamento das classes a serem utilizadas no sistema. Um sistema OO é composto por classes e por um conjunto de objetos que colaboram ou interagem para execução dos serviços (casos de uso) oferecidos pelo sistema. Os componentes estruturais do sistema são as classes. Como não existe uma forma sistematizada de levantá-las, esta tarefa baseia-se em experiência e criatividade (STADSZISZ, 2002).

Ainda durante esta fase, foram estudados os possíveis relacionamentos entre as classes, o que deu origem a diferentes diagramas referentes a cada pacote. Verificou-se que há classes de diferentes pacotes que se inter-relacionam. Também para esta finalidade foi utilizada a ferramenta Together.

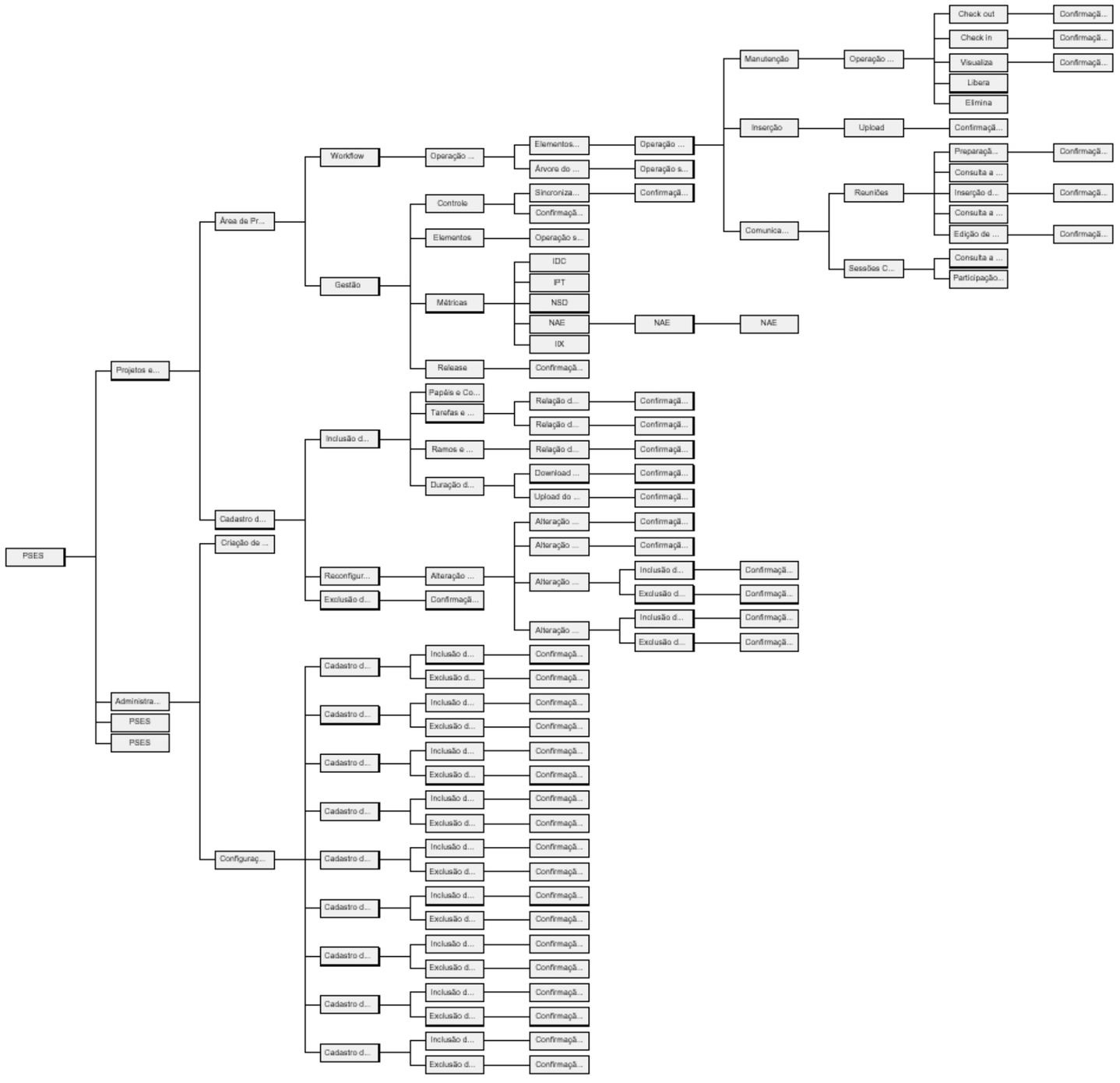
Os diagramas elaborados neste momento da implementação não foram aproveitados diretamente na codificação do sistema, em função de ter-se optado por uma arquitetura mais simples, baseada em JSP e JavaBeans (KURNIAWAN, 2002). Caso a opção tivesse sido a utilização da especificação J2EE e EJBs, tais diagramas poderiam ter sido aproveitados diretamente. O Apêndice 8 traz o estudo de relacionamento entre classes referente ao modo de administração; o Apêndice 9 traz o diagrama de classes referente ao modo de gestão e; o Apêndice 10 traz o diagrama de classes referente ao modo de operação do sistema.

Na seqüência, foi criado um modelo de interface e hierarquia de páginas HTML que posteriormente foi aproveitado quase que diretamente no sistema. A finalidade deste estudo foi a de definir a diagramação das páginas do sistema e os mecanismos de navegação. Também através deste modelo, muitas funcionalidades desejáveis foram detectadas e incorporadas aos modelos de casos de uso.

Para a tarefa de diagramação das páginas HTML, foi utilizada a ferramenta **Microsoft FrontPage 98** (MICROSOFT CORPORATION, 1998). Através dela foi criado um *design* simples, cujo padrão pudesse ser replicado todas as páginas do sistema e um esboço da estrutura em linguagem HTML que permitisse a navegação pelas páginas. A possibilidade de publicação das páginas HTML diretamente para o diretório de trabalho da ferramenta IDE de desenvolvimento também se mostrou

bastante interessante, num processo produtivo de criação de páginas e conversão das mesmas em páginas de conteúdo dinâmico. Desta forma, o trabalho de criação das páginas pôde ser suficientemente isolado do trabalho de criação da lógica do sistema. A Figura 16 apresenta uma visão da hierarquia das páginas do esboço da estrutura.

Figura 16 - Hierarquia de páginas HTML



3.4.6 Codificação do sistema

A etapa de codificação do sistema PSES (Plataforma de Suporte à Engenharia Simultânea) se estendeu de março de 2002 a abril de 2003. O sistema é composto por três camadas lógicas: uma camada de apresentação, uma camada de lógica de negociação, e uma camada de dados (ROMAN, AMBLER e JEWELL, 2002). Na camada de apresentação ficam as partes que tratam da interação com o usuário através de uma interface. A camada de lógica de negociação contém componentes que tratam da lógica de programação da aplicação. Finalmente, a camada de dados é utilizada pela camada de lógica de negociação para armazenar seu estado de forma permanente. Esta última consiste de um banco de dados relacional que contém arquivos de projeto do produto. Para esta atividade não foi utilizada a possibilidade de armazenagem de dados através de formato XML, prática esta que ocorre com freqüência cada vez maior nos sistemas atuais.

Dentre as tecnologias disponíveis hoje para o desenvolvimento de sistemas destinados ao ambiente *Web*, similares ao sistema proposto nesta investigação, ***servlets***[15], **JSP** e **JavaBeans** foram escolhidos. São tecnologias amplamente aceitas na indústria, por terem sido bem fundamentadas, estabelecidas, comprovadas, documentadas e difundidas (HALL, 2000). Além disso, utilizam linguagem Java, o que facilita seu uso, uma vez que não se trata de uma outra linguagem (tal como CGI-Perl ou PHP) cuja sintaxe deva ser adicionalmente assimilada.

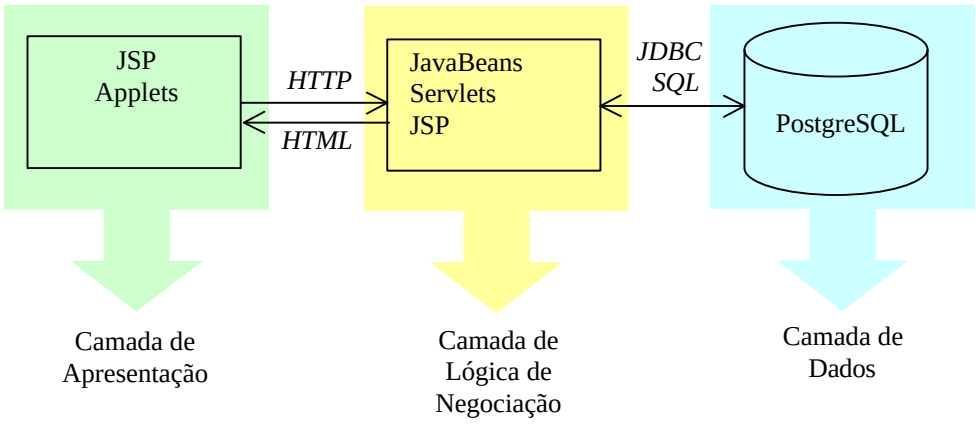
O sistema utiliza como princípio básico de funcionamento a atualização dinâmica de páginas HTML baseadas em resultados de consultas a um banco de dados

---

[15] *Servlet* é uma classe Java que pode ser carregada dinamicamente e executada por um servidor *web* específico denominado "contêiner *servlet*", ou "contêiner *web*" no caso do servidor ser também um "contêiner JSP" (KURNIAWAN, 2002, p. 6). Servidores *web* comuns, tais como o Apache, permitem que apenas páginas HTML estáticas sejam servidas.

e no processamento lógico destas informações. Para tanto, utiliza-se a tecnologia JDBC, que permite o acesso a bancos de dados relacionais através de programas escritos em Java, e comandos SQL. O protocolo HTTP é utilizado para converter a solicitação do usuário feita através da interface em informações a serem processadas na camada de lógica de negociação. Uma vez capturada a solicitação, ocorrem consultas ao banco de dados, cujos resultados são processados e convertidos em linguagem HTML para apresentação final ao usuário.

Figura 17 - Arquitetura e tecnologias do sistema proposto

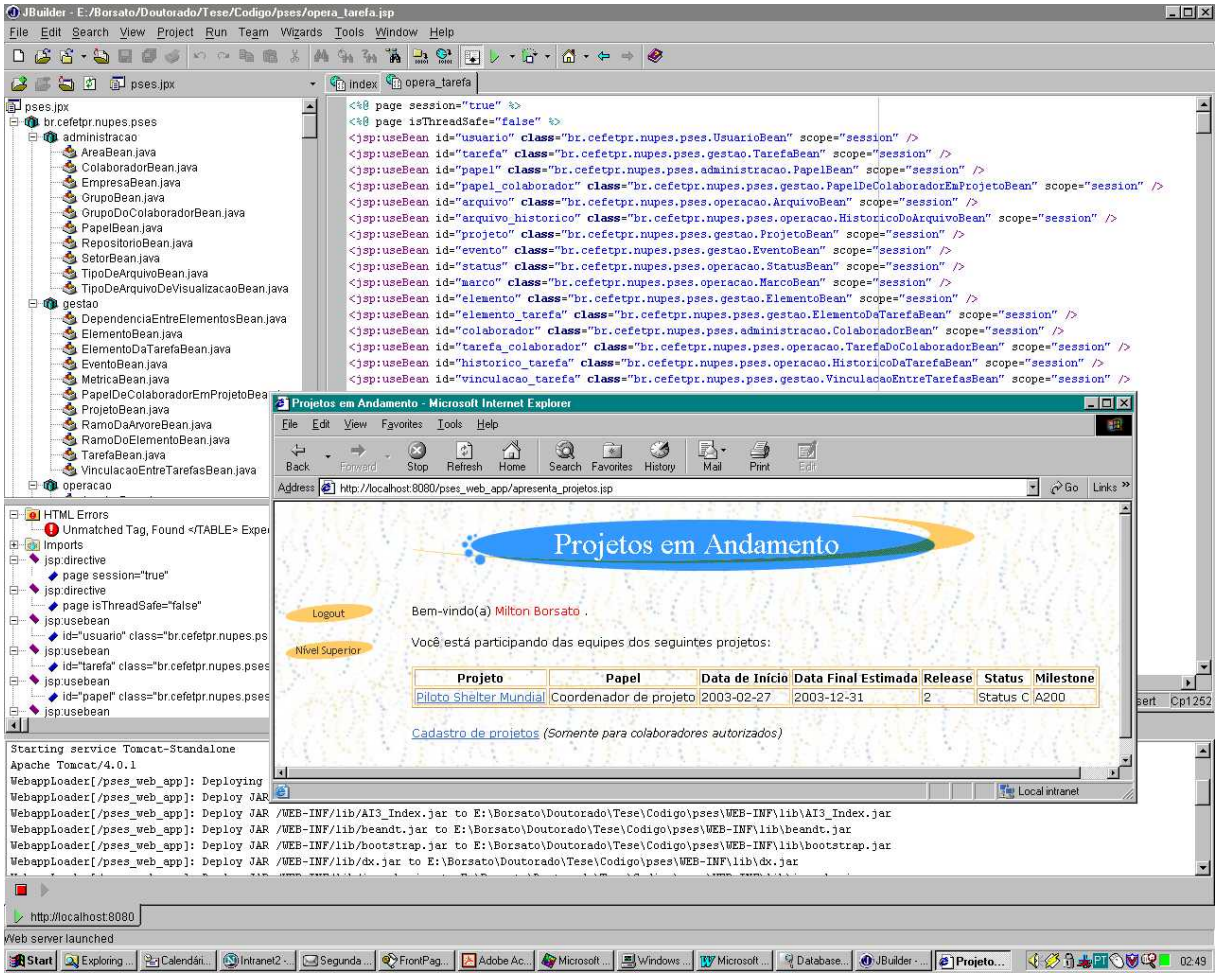


O sistema ainda faz uso de protocolo FTP para transferência de arquivos entre repositórios distribuídos e o posto local de trabalho do usuário, além de recursos de *e-mail* para notificação de eventos e *applets* Java para funções específicas que operam no lado cliente do sistema, tais como aqueles utilizados para visualização de *workflow*, *upload* de arquivos ao servidor e visualização da árvore de decomposição do produto. Para algumas operações de atualização imediata das páginas em resposta a determinadas opções selecionadas pelo usuário, foram utilizados recursos de JavaScript.

Ao desenvolver aplicativos *Web* em Java, há duas principais arquiteturas que podem ser adotadas (KURNIAWAN, 2002). A primeira utiliza *servlets*, JSPs e JavaBeans na camada de lógica de negociação. Normalmente este modelo de projeto é adequado a aplicativos de porte pequeno e médio. A segunda inclui o uso da

especificação J2EE através de um servidor J2EE e Enterprise JavaBeans (EJB), e é especialmente útil para aplicações que necessitam de escalabilidade. Nesta pesquisa optou-se pela primeira arquitetura. A Figura 17 apresenta a arquitetura e tecnologias escolhidas.

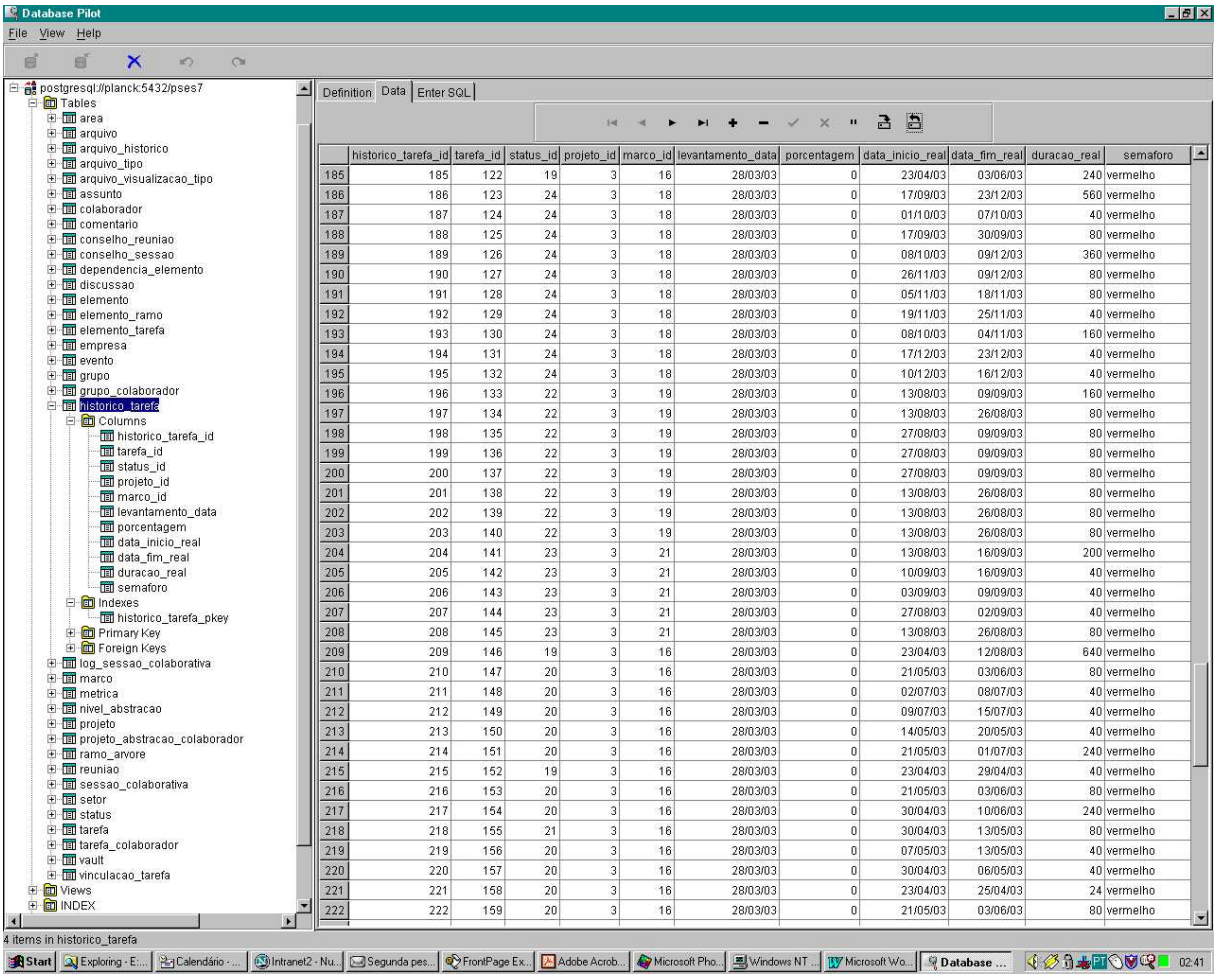
Figura 18 - Visão do processo de desenvolvimento de código Java utilizando a ferramenta JBuilder6



Uma vez escolhidas arquitetura e tecnologia a serem utilizadas, partiu-se para a escolha de uma ferramenta IDE de desenvolvimento Java de *applets*, JSPs, *servlets* e JavaBeans, e dos aplicativos servidores de *servlets* e páginas HTML/JSP. Em função dos recursos que dispõe e facilidade de aprendizado, a ferramenta **JBuilder** (BORLAND SOFTWARE CORPORATION, 2000) foi escolhida. Inicialmente, a versão 4 foi adotada. Nesta, encontra-se incorporado o contêiner *Web*

Tomcat versão 3.1 (APACHE SOFTWARE FOUNDATION, 1999), o que facilita imensamente o trabalho de desenvolvimento, pois não há necessidade de haver uma instalação de servidor à parte. No caso de necessidade de reinicialização do servidor, a operação pode ser feita diretamente da ferramenta IDE. É a configuração mais indicada para o desenvolvimento de aplicativos *Web*. Posteriormente, a versão 6 da ferramenta **JBuilder** foi adotada (BORLAND SOFTWARE CORPORATION, 2002a), por oferecer maior estabilidade, melhores recursos e por trabalhar com a versão 4.0.1 do servidor **Tomcat**. A Figura 18 apresenta uma visão geral da interface da ferramenta JBuilder versão 6.

Figura 19 - Tabelas do banco de dados PostgreSQL visualizadas através da ferramenta Database Pilot



O JBuilder versão 6 oferece ainda muitos outros recursos úteis, tais como

visualização direta de dados do banco, documentação do sistema no padrão Javadoc, visualização de classes e relacionamentos no padrão UML, entre outros. A Figura 19 ilustra como o banco de dados pode ser visualizado através da ferramenta **Database Pilot** incorporada ao JBuilder versão 6.

Uma vez caracterizada esta pesquisa e descritos os métodos e ferramentas utilizadas para especificação, modelagem e implementação do código referente ao sistema proposto, seguem os resultados obtidos e discussão sobre os mesmos.

4 RESULTADOS

4.1 O SISTEMA PSES

O sistema desenvolvido nesta pesquisa foi denominado PSES (Plataforma de Suporte à Engenharia Simultânea). Este permite a criação de projetos baseados em cenários. Cada cenário é, na verdade, um modelo de referência segundo o qual um produto de determinada natureza deve ser desenvolvido por uma equipe interdisciplinar. Compreende um modelo de processos e um modelo de informações, os quais devem ser levantados *a priori*.

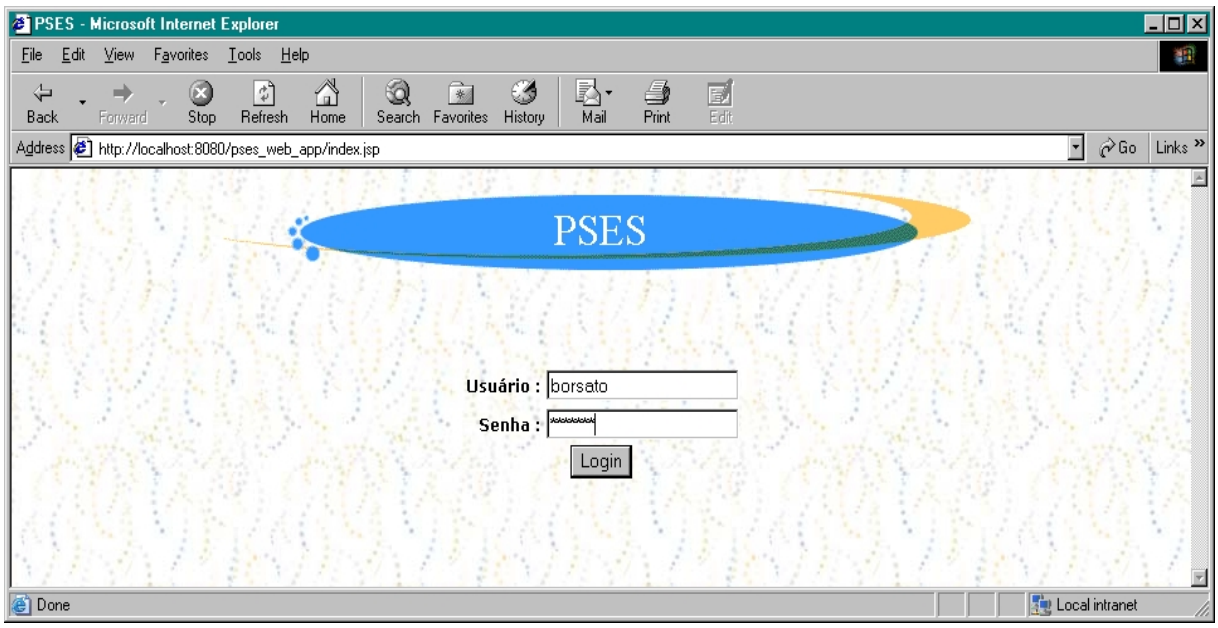
Não há restrições com relação à natureza do produto a ser desenvolvido, o que permite que produtos que muitas vezes não são entendidos como artefatos, também possam ser contemplados pela plataforma, tais como *software* e produtos químicos. Para que se pudesse focar no sistema como prova de conceito para validar os requisitos especificados para o mesmo, não foram desenvolvidos mecanismos específicos que permitissem a inserção de cenários de forma sistematizada pelo administrador do sistema. Isto, contudo, não impede que tais mecanismos sejam inseridos posteriormente.

Os dois cenários que serviram como ponto de partida para a elaboração do sistema, enquanto prova de conceito, foram alimentados manualmente ao sistema, através de *scripts* SQL, inserção de JSPs específicos relativos ao *workflow* e cronograma mestre em formato XML compatível com o padrão adotado pelo Microsoft Project 2002. Como ocorre a necessidade de inserção de JSPs a cada novo cenário inserido, há a necessidade de recompilação do aplicativo e respectivo *deployment*[16].

[16] *Deployment* é o processo de instalação para uso de um aplicativo W*eb* num servidor do tipo "contêiner *web*".

As diversas funcionalidades do sistema PSES podem ser acessadas conforme o grupo ao qual pertença o usuário. Usuários do grupo **Administração** têm acesso aos mecanismos de cadastramento de usuários, empresas, setores, áreas profissionais, entre outros. Usuários do grupo **Project managers** têm acesso às facilidades para criação e reconfiguração de projetos e outras funcionalidades específicas, tais como acesso a métricas e atualização de documentos-mestres do projeto (*eg.* cronograma). Os demais usuários têm direito de alteração e inserção restrito às suas responsabilidades nos projetos em que participam, mas podem acessar as demais áreas de dados para consulta. A Figura 20 traz a tela de *logon* do sistema PSES.

Figura 20 - Tela de *logon* do sistema PSES

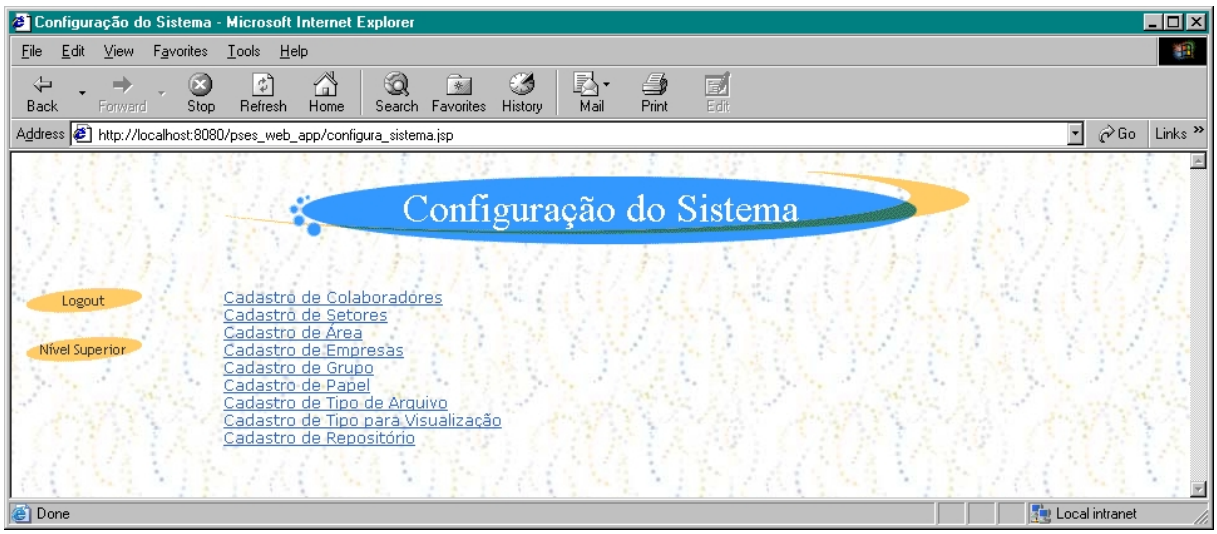


4.2 MODO DE ADMINISTRAÇÃO

O modo de administração é o primeiro dos três casos de uso para o sistema, conforme ilustrado na Figura 12. Caso o usuário seja administrador, as opções de operação são apenas aquelas referentes ao cadastramento de itens. O administrador

pode cadastrar empresas, setores de empresas, colaboradores, áreas profissionais[17], papéis[18] a serem desempenhados nos projetos, tipos de arquivo a serem armazenados, tipos de arquivo para visualização e repositórios a serem utilizados para armazenagem dos arquivos. A Figura 21 apresenta as opções de cadastramento para o administrador do sistema. Em seguida, o administrador opta pela inclusão ou exclusão do item.

Figura 21 - Opções de cadastramento para a administração do sistema



Para garantir a consistência das informações cadastrais, foram inseridas funções em JavaScript que permitem que as seleções guiem sub-seleções e assim por diante. Desta forma, uma vez selecionada uma empresa, apenas os setores daquela empresa são apresentados como opções. A Figura 22 apresenta as opções de setores na empresa escolhida para o caso de inclusão de um novo colaborador.

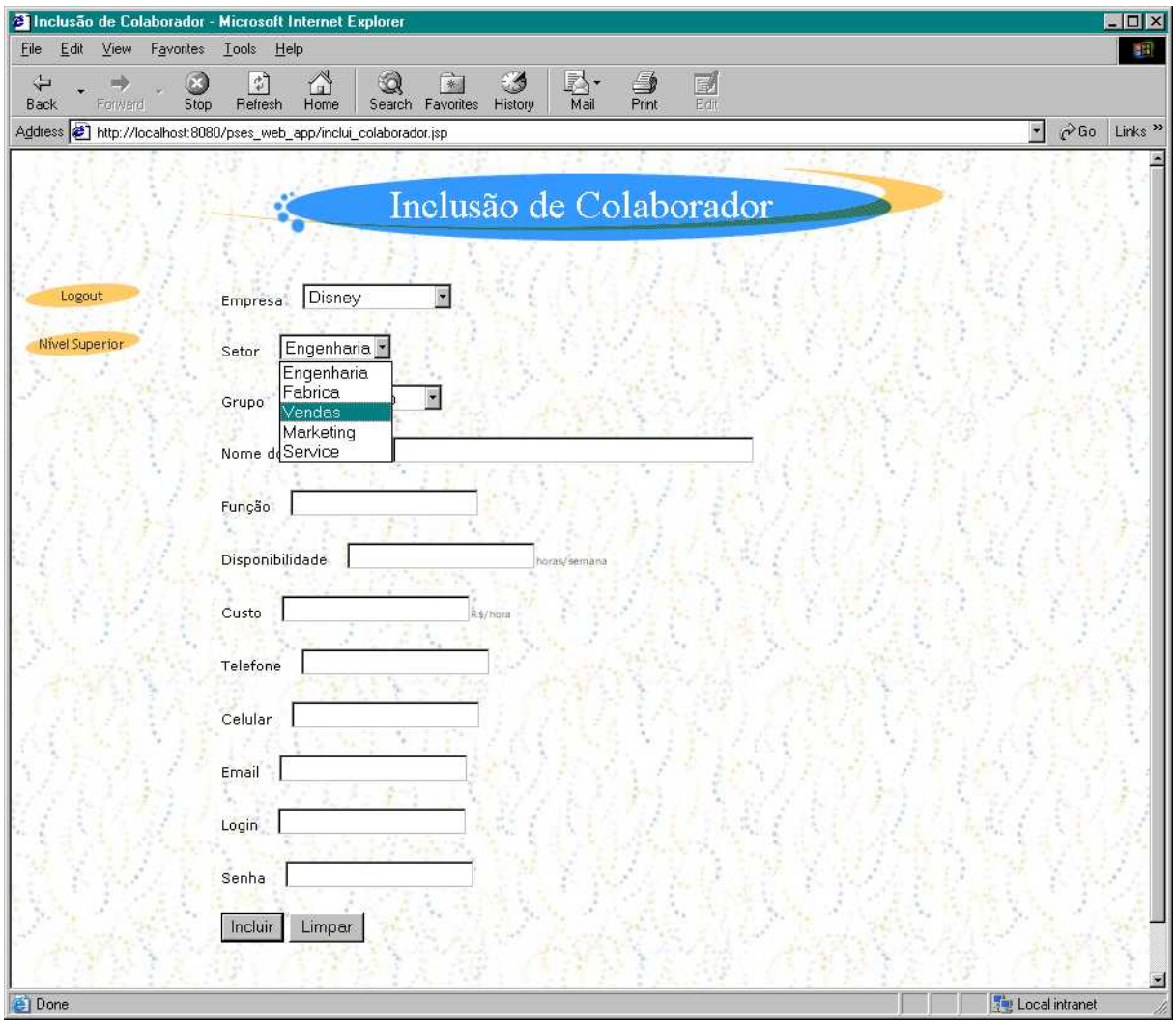
Similarmente, é possível excluir itens cadastrados. Entretanto, itens utilizados em projetos não podem ser excluídos sem que os projetos que os utilizam sejam eliminados primeiramente. Também neste caso foram utilizadas funções em JavaScript para orientar as seleções. No caso de exclusão de colaborador, por exemplo, apenas os colaboradores do setor e da empresa selecionados são listados para a seleção

[17] As áreas profissionais são utilizadas para cálculo de métricas

[18] Ss papéis são os níveis de abstração segundo os quais os colaboradores observam o projeto.

final. A Figura 23 apresenta esta situação.

Figura 22 - Cadastramento de colaboradores: seleção de setor de empresa

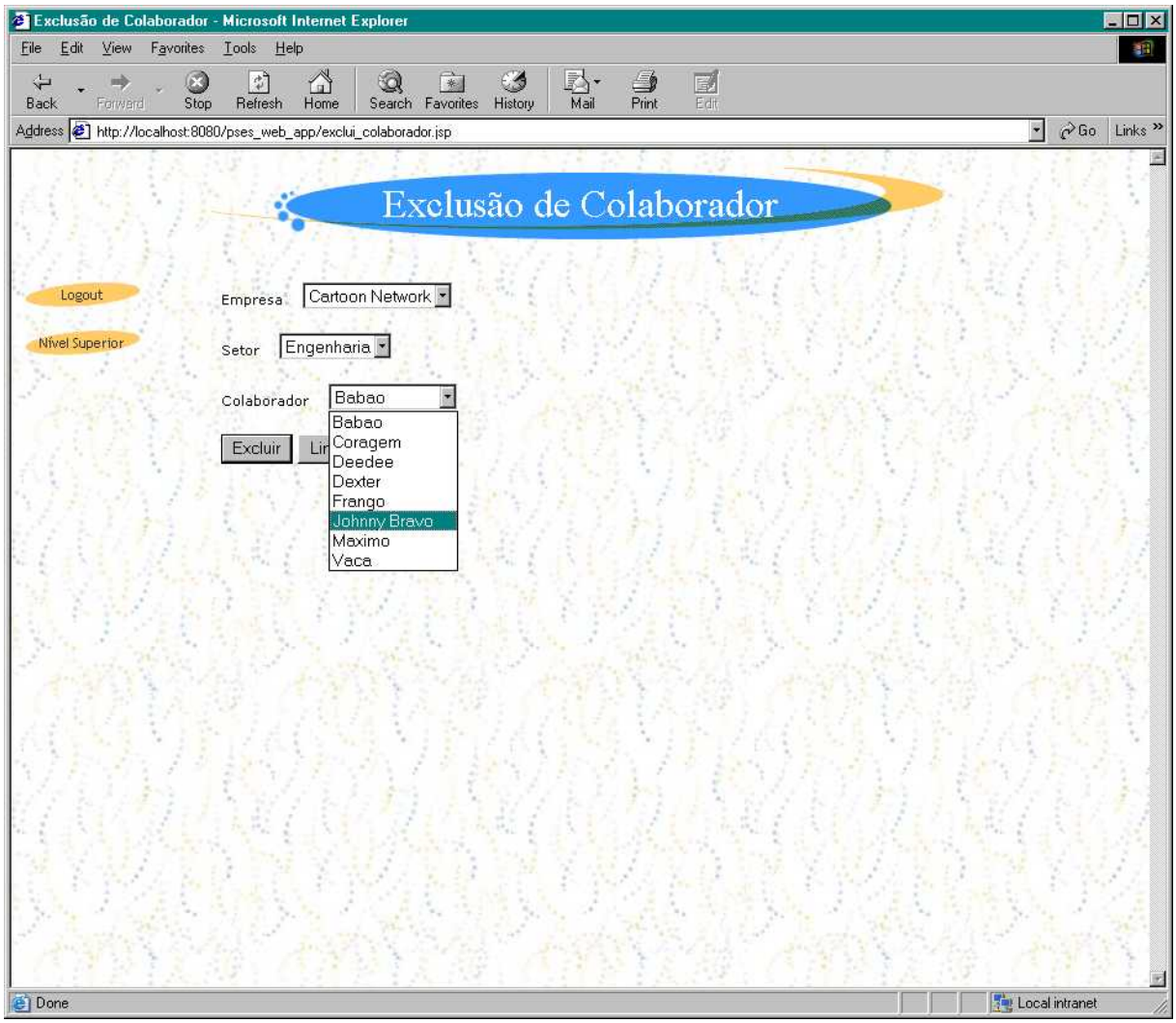


No sistema PSES, assim como em sistemas que executam funções de PDM, as informações de projeto são armazenadas em arquivos, que por sua vez compõem um elemento de informação. Portanto, um elemento de informação poderá conter diversos arquivos, cada qual de um tipo. Por exemplo, podem ser arquivos gerados por ferramentas CAD ou CAE, ou mesmo textos e imagens que sejam úteis de alguma forma à equipe que desenvolve o projeto.

Entretanto, não são os arquivos, mas sim os elementos de informação que guardam relações de dependência entre si. E, cada tarefa do *workflow* pode trabalhar

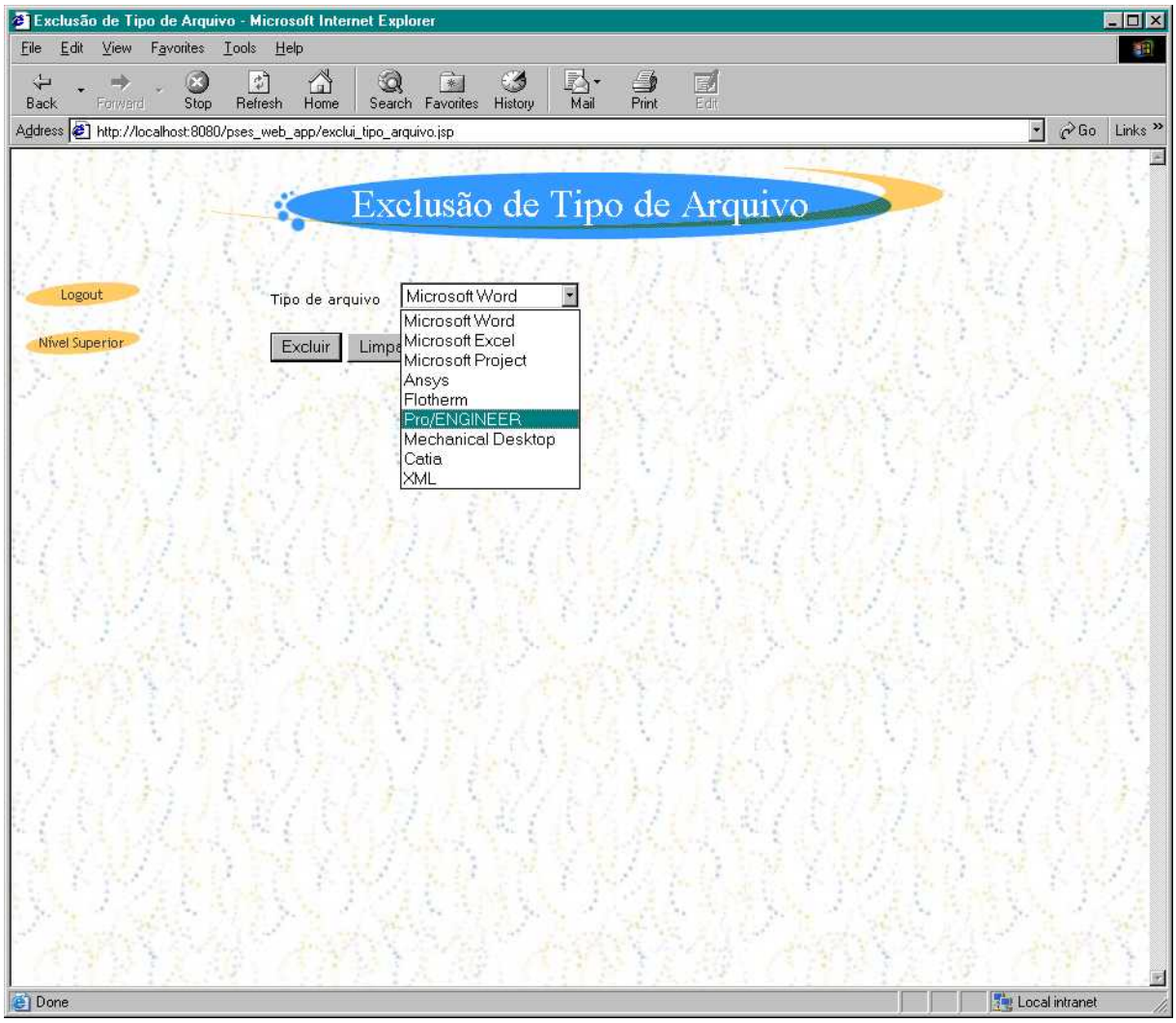
quantos elementos de informação forem definidos. A eles podem ser adicionados novos arquivos, conforme o andamento do projeto.

Figura 23 - Seleção de colaborador para exclusão



Cabe ao administrador do sistema cadastrar os tipos de arquivos que serão armazenados. O sistema não verifica se realmente um arquivo inserido, por exemplo, é do tipo especificado no processo de inserção. Entretanto, esta informação é útil pois qualquer colaborador da equipe pode solicitar uma cópia do arquivo e manipulá-la se tiver disponível o aplicativo que o gerou. Só não poderá atualizá-lo ou eliminá-lo, se não for o responsável pelo mesmo. A Figura 24 apresenta a tela de exclusão de tipos de arquivo.

Figura 24 - Exclusão de tipos de arquivo

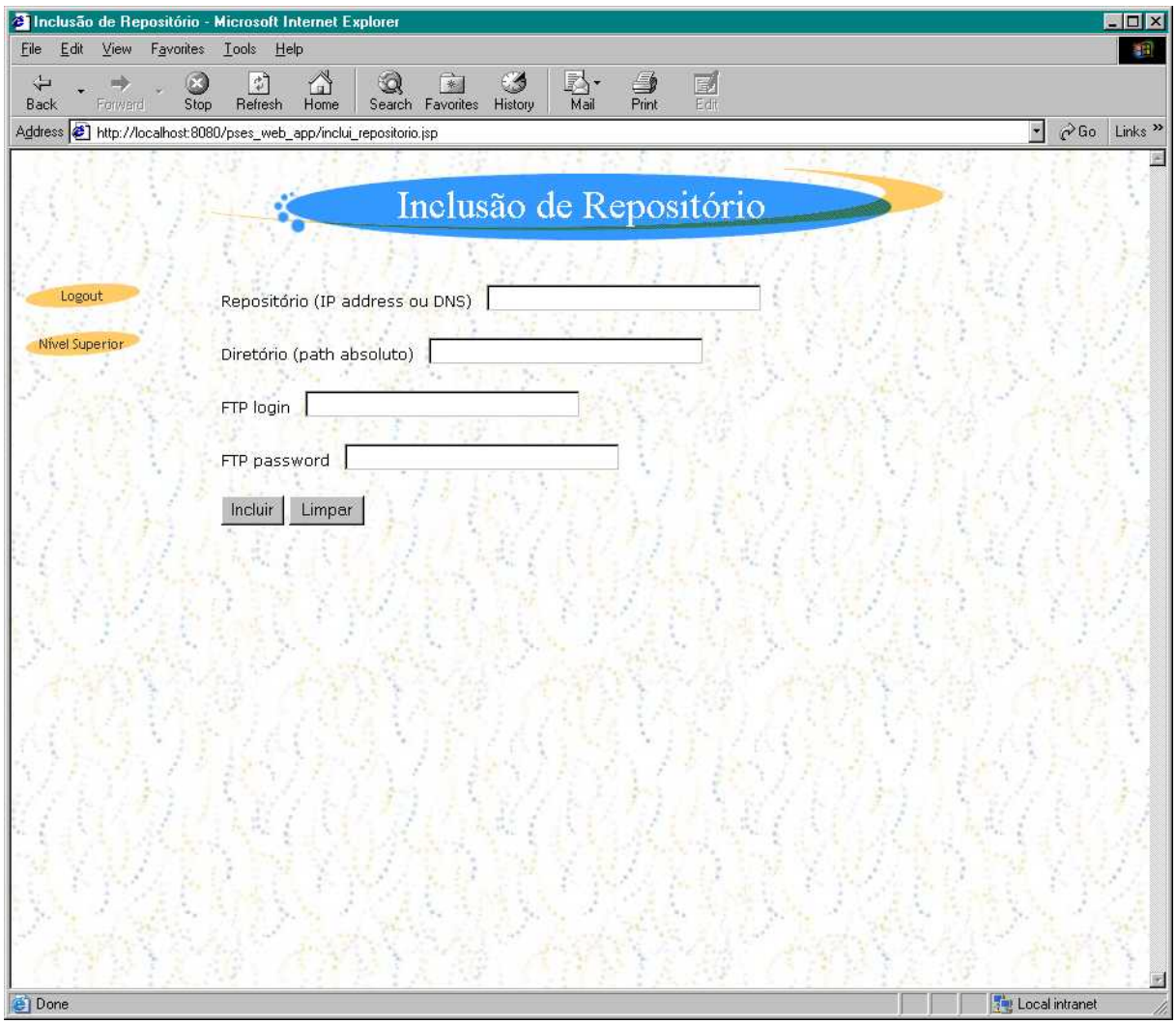


De maneira análoga, o administrador do sistema deverá cadastrar tipos de arquivo para visualização. Cada arquivo de um elemento de informação, no momento em que é inserido ou atualizado deve ser acompanhado de um outro que traga uma representação em formato facilmente visualizável pelos companheiros de equipe. Por exemplo, um arquivo no formato Microsoft Word poderá ser visualizado desde que seu arquivo correspondente de visualização tenha sido inserido em formato PDF.

Normalmente estes formatos de visualização possuem *plug-ins* específicos gratuitos que podem ser instalados nos navegadores. Assim, colaboradores que estejam trabalhando em estações de trabalho que não disponham do aplicativo utilizado para gerar o documento, poderão visualizar o arquivo PDF no próprio navegador. Existem

inúmeras opções de formatos de visualização, desde as mais simples, como PDF, GIF e JPG, até as mais sofisticadas, como MPEG-2 e VRML (ANUPAM e BAJAJ, 1994). Obviamente, a geração dos arquivos de visualização deverá ser feita pelo colaborador responsável pelo arquivo fonte, e deverá escolher a melhor forma de representar aquela informação. Outra vantagem deste sistema é que arquivos de visualização são normalmente menores que os arquivos fonte. Assim, os colaboradores terão menos dificuldades para terem acesso às informações de projeto através da rede.

Figura 25 - Cadastramento de repositórios



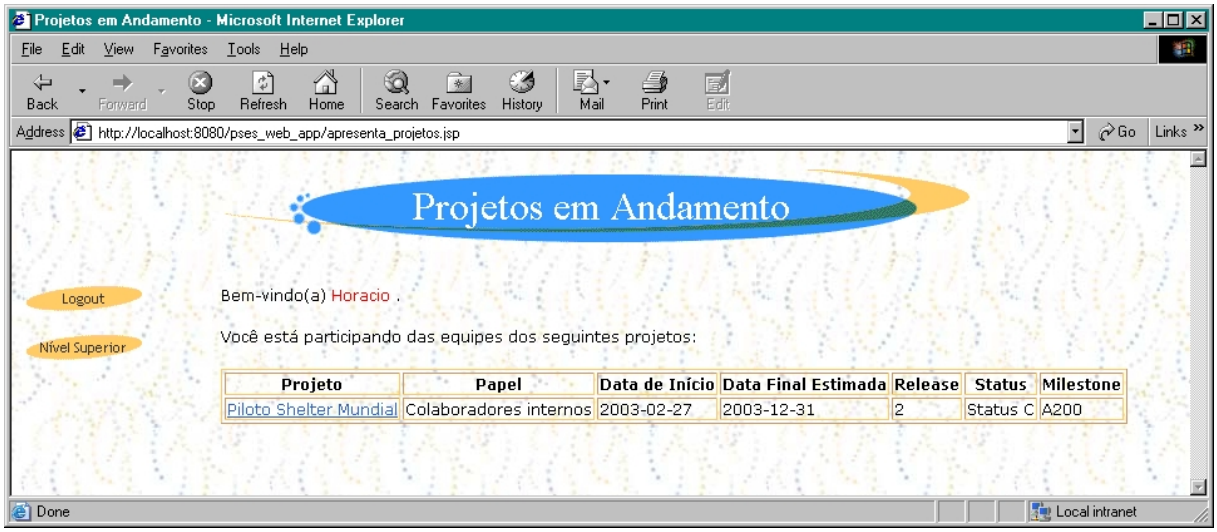
Na prática, pretende-se distribuir os ramos da árvore de decomposição e respectivos elementos e arquivos de forma que fiquem mais próximos dos usuários que

os criaram, para melhorar o desempenho do sistema. Ao mesmo tempo, os demais integrantes da equipe de desenvolvimento têm acesso aos arquivos independentemente de sua localização. Cabe ao administrador de sistemas cadastrar os repositórios para que os gerentes de projeto possam utilizá-los quando da configuração dos projetos. A Figura 25 apresenta o cadastramento de repositórios.

4.3 MODO DE OPERAÇÃO

O modo de operação é o segundo dos três casos de uso para o sistema, conforme ilustra a Figura 12. Este modo é ativado após o *logon* do usuário. Uma vez consultado o banco de dados, caso não se trate do administrador do sistema, o usuário é informado de quais projetos participa e a situação de cada um deles, com relação às datas de início e término previstas, status do modelo de informação e último *milestone* realizado. A Figura 26 apresenta esta tela.

Figura 26 - Tela de apresentação de projetos

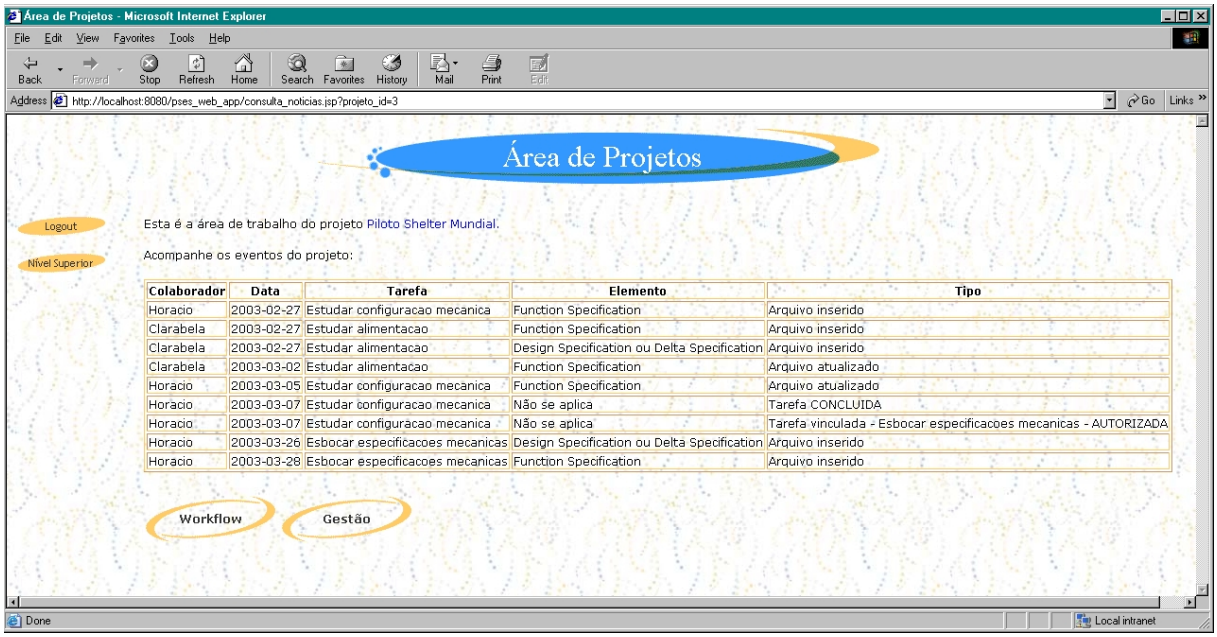


Uma vez escolhido o projeto no qual pretende trabalhar, o colaborador membro da equipe de desenvolvimento será informado dos eventos que ocorreram desde a criação do projeto. A Figura 27 apresenta esta operação.

Eventos são operações de inserção, atualização e visualização de arquivos,

além de conclusão e autorização para iniciar tarefas. Também, pode haver eventos criados para promover reuniões colaborativas. Assim pretende-se criar uma maneira de informar a respeito do andamento do projeto de forma automatizada. O colaborador pode então iniciar seu trabalho selecionando o projeto que desejar.

Figura 27 - Notícias do projeto



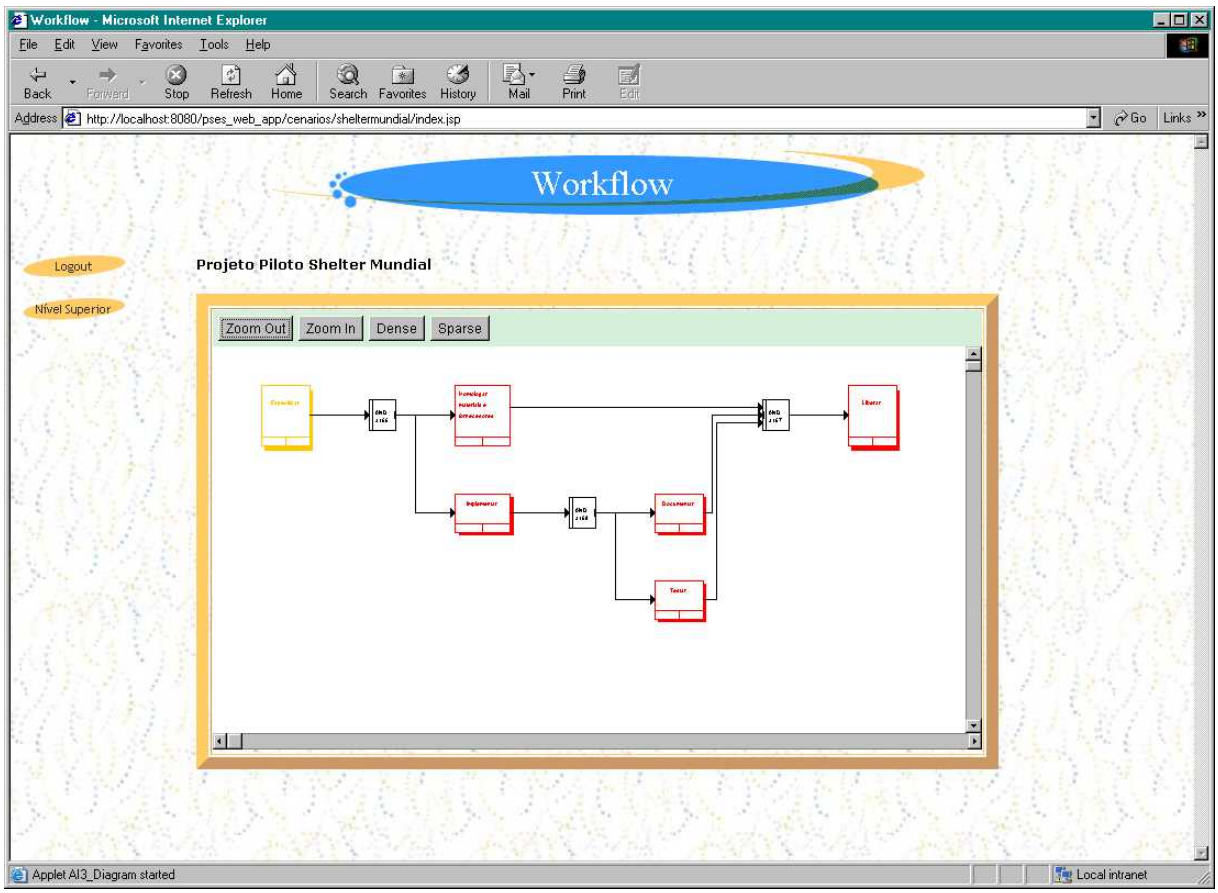
A entrada num projeto cadastrado no sistema PSES sempre se dá através do *workflow.* Portanto o colaborador deve conhecer as tarefas para as quais foi designado. Caso contrário, ao escolher uma tarefa sobre a qual não responde, não poderá promover alterações dos arquivos dos elementos de informação vinculadas a ela, nem alterar o seu registro de progresso. Poderá, entretanto, visualizar arquivos e verificar o progresso das atividades. A Figura 28 apresenta a visão mais geral do *workflow*, através da qual o colaborador pode navegar. Os botões superiores permitem um ajuste da representação gráfica para melhor visualização dos itens.

Conforme estabelecido no padrão IDEF3, os retângulos sombreados são tarefas compostas. Para observar o seu conteúdo, o colaborador as seleciona e é conduzido a um nível inferior do fluxo. A Figura 29 mostra parte de uma decomposição de tarefa.

As cores dos retângulos são indicativas do progresso das mesmas. A cor vermelha indica que a tarefa não pode ser iniciada porque as tarefas das quais depende não foram concluídas ainda. A cor amarela indica que a tarefa está em andamento. A cor azul indica que a tarefa já foi concluída.

Ao atingir a tarefa na qual pretende trabalhar, o colaborador a escolhe para acessar as informações de seu histórico, eventos relacionados à mesma e um menu onde pode atualizar o progresso da atividade. A Figura 30 traz detalhes desta apresentação.

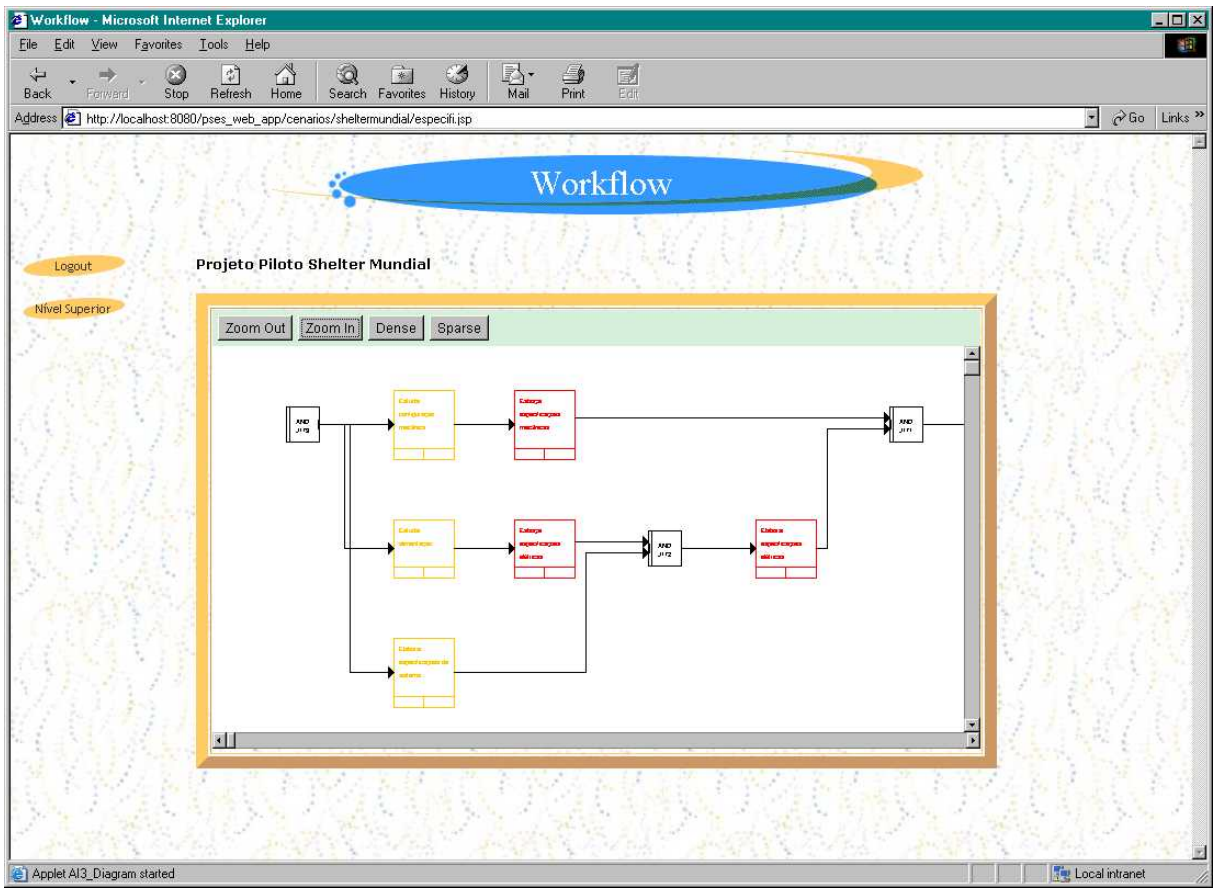
Figura 28 - Visão de topo do *workflow*



O colaborador pode, neste instante, atualizar o progresso da tarefa em questão. Neste caso um novo registro de histórico da tarefa é criado. Percentuais inferiores aos correntes não são apresentados como opção. Ao atualizar o progresso da tarefa, a página é atualizada dinamicamente para fazer constar o registro inserido.

Se o percentual for ajustado para 100%, uma série de verificações é feita de forma transparente ao usuário. Primeiramente, o semáforo da tarefa em questão é atualizado para azul, o que significa que ela foi completada. Em seguida, um **evento** é gerado e, caso algum colaborador entre no sistema, a consulta a notícias do projeto apresenta o mesmo numa tabela (ver Figura 27).

Figura 29 - Decomposição de uma tarefa



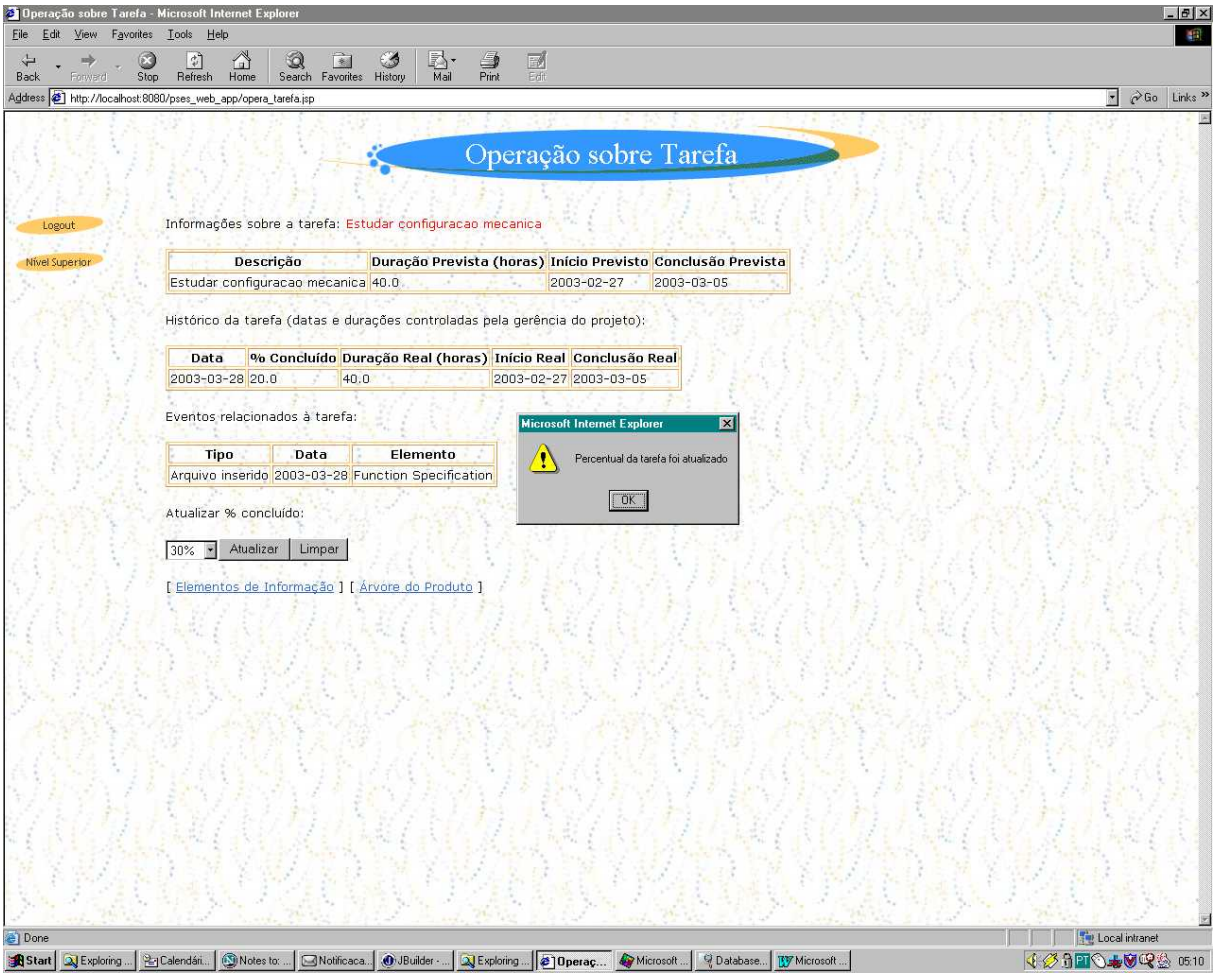
Como parte do mecanismo de notificação automática do sistema, o gerente do projeto é, na seqüência, informado a respeito da conclusão da tarefa via *e-mail*, conforme ilustrado na Figura 31. O servidor de *e-mail* utilizado neste procedimento é aquele instalado na máquina que contém o ramo "tronco" do projeto.

A conclusão de uma tarefa pode ocasionar a alteração do semáforo de outras vinculadas a ela. A Figura 32 apresenta uma situação na qual uma alteração da tarefa "Esboçar especificações mecânicas", cujo semáforo passa de amarelo para azul

provoca a autorização da tarefa "Elaborar especificações mecânicas", que por sua vez dependia da conclusão de mais de uma tarefa. Neste caso o semáforo da tarefa vinculada é alterado para amarelo e ocorrem automaticamente notificações por *e-mail* e geração de eventos. Se o colaborador a ser informado coincidir com o informante, a mensagem não é enviada. A Figura 33 mostra um *e-mail* autorizando o início de uma tarefa vinculada.

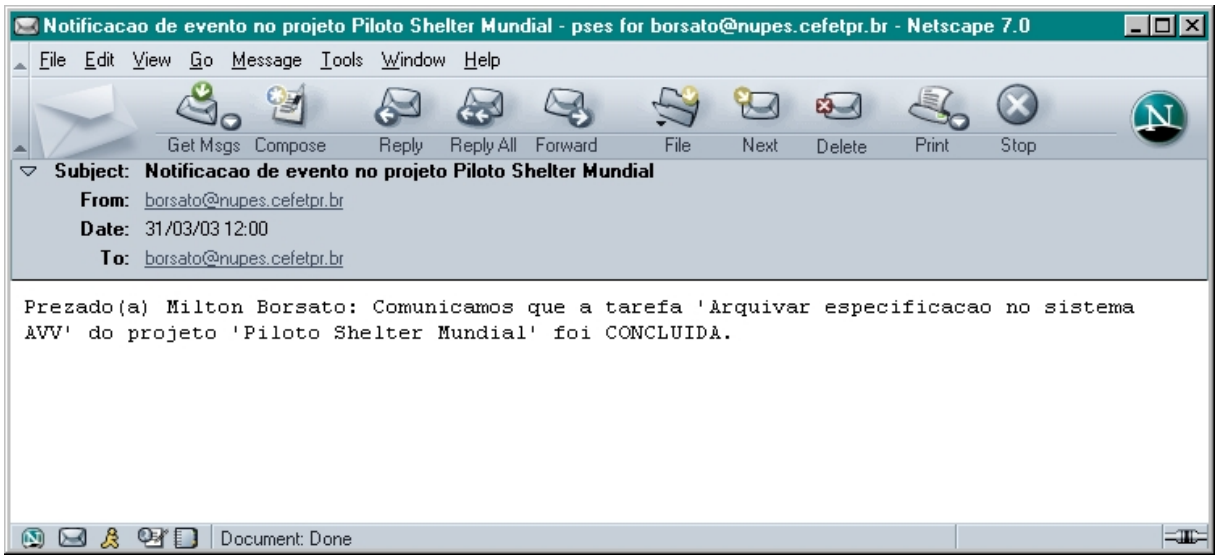
Da mesma forma, a conclusão de uma tarefa pode ocasionar a conclusão de uma tarefa-mãe (composta), caso todas as tarefas-filha tiverem sido completadas. Neste caso, também ocorre uma verificação das tarefas vinculadas à tarefa-mãe, para autorizar o seu início. Novamente, *e-mails* notificadores e eventos são gerados automaticamente pelo sistema.

Figura 30 - Situação da tarefa



O status de um projeto depende basicamente da conclusão de um conjunto de tarefas que manipulam determinadas informações (ver 3.4.2). Portanto a cada conclusão de tarefas, ocorre uma verificação se o projeto pode avançar de status. Cada tarefa guarda, como informação no banco de dados, o status das informações que manipula. No caso de uma atualização de status, ocorre notificação por *e-mail* do gerente do projeto e um evento é gerado. De forma análoga, os marcos de um projeto são verificados a cada conclusão de tarefa. Status e marco são atualizados nos respectivos campos da tabela `projeto` do banco de dados.

Figura 31 - *E-mail* notificando a conclusão de uma tarefa



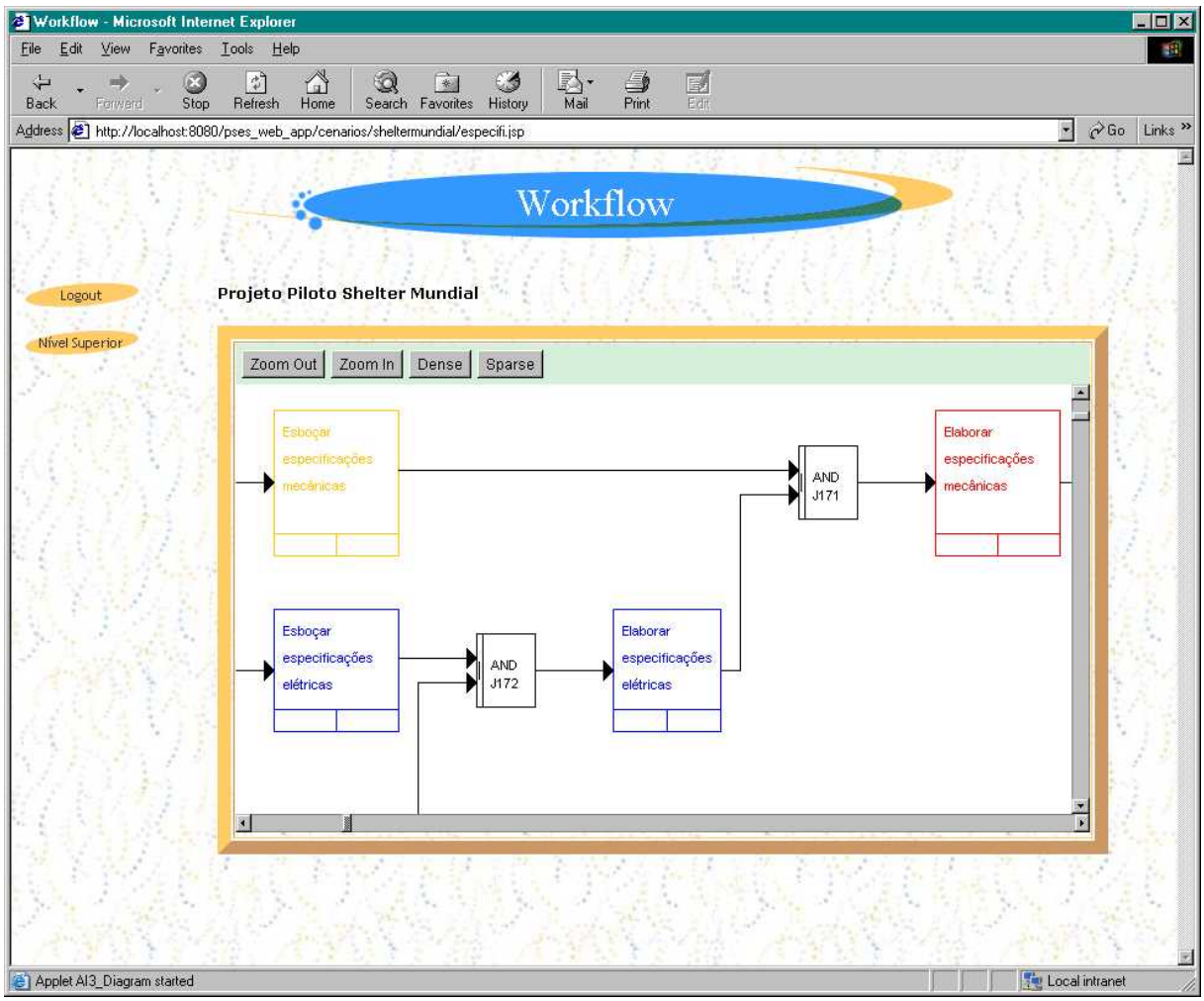
Conforme apresentado na Figura 30, o colaborador pode optar pela apresentação dos elementos de informação referentes à tarefa em questão, ou observar os elementos via árvore de decomposição do produto. Neste último caso, se o colaborador escolher arquivos de elementos sobre os quais não tem direito de modificação, a permissão é negada pelo sistema.

O acesso via árvore de decomposição do produto tem a finalidade de facilitar a busca por informações sobre as quais outros colaboradores tenham a responsabilidade. Como nem é difícil localizar exatamente a tarefa que manipule tal elemento, é mais simples que a localização seja feita através da árvore. A Figura 34

apresenta a possibilidade de localização de elementos desta forma.

Uma vez escolhido o meio de acesso aos elementos de informação, os mesmos são apresentados ao usuário, conforme ilustra a Figura 35. Apenas a seleção de um elemento de informação por vez é permitida, através do recurso de *radio button* em HTML.

Figura 32 - Situação de atualização de tarefas vinculadas



Na seqüência, o usuário deve indicar que tipo de operação será efetuada no elemento de informação escolhido: **manutenção**, **inserção** ou **comunicação**, conforme ilustra a Figura 36. A inserção corresponde ao *upload* de um arquivo ainda não constante da base de dados. A manutenção compreende operações de *check-in*, *check-out*, visualização, liberação e eliminação de arquivos, a serem escolhidas na

seqüência. A comunicação permite que o colaborador agende, consulte e participe de reuniões e sessões colaborativas.

Figura 33 - *E-mail* notificando a autorização de uma tarefa ao responsável

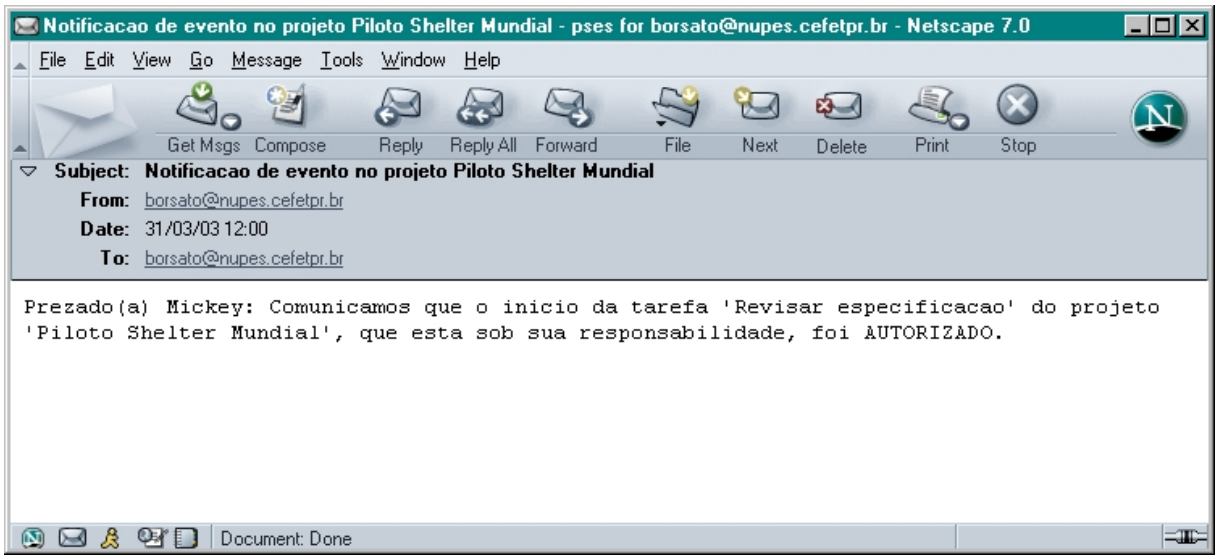


Figura 34 - Localização de elementos via árvore de decomposição do produto

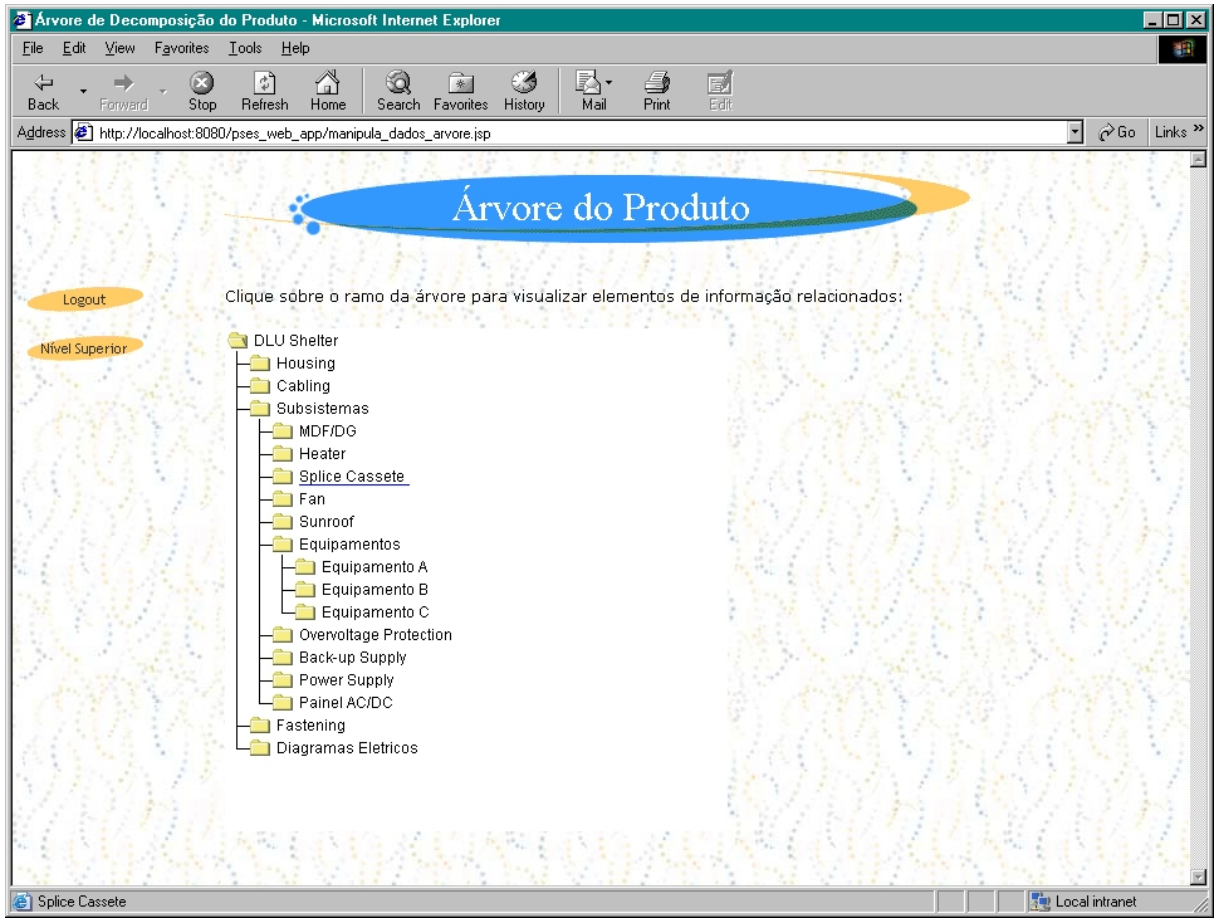


Figura 35 - Apresentação de elementos de informação

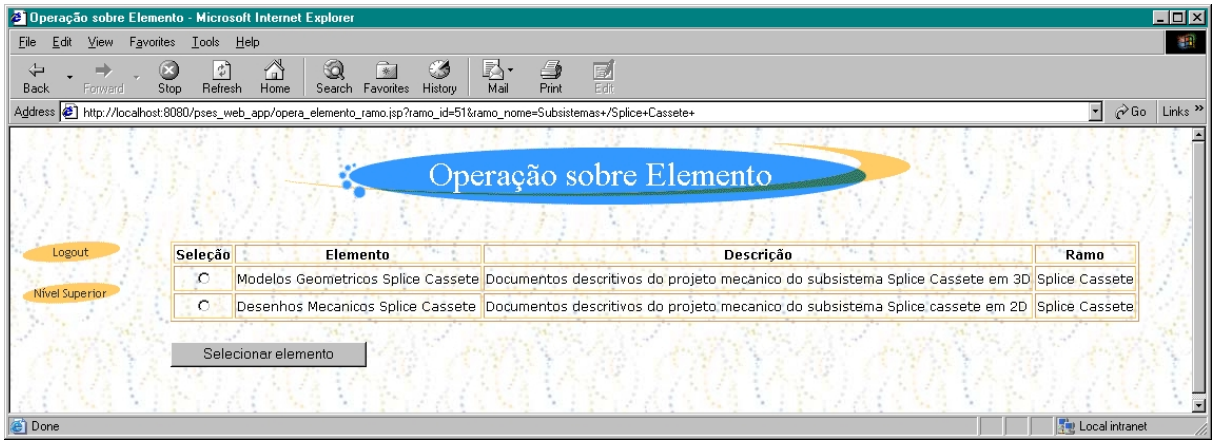
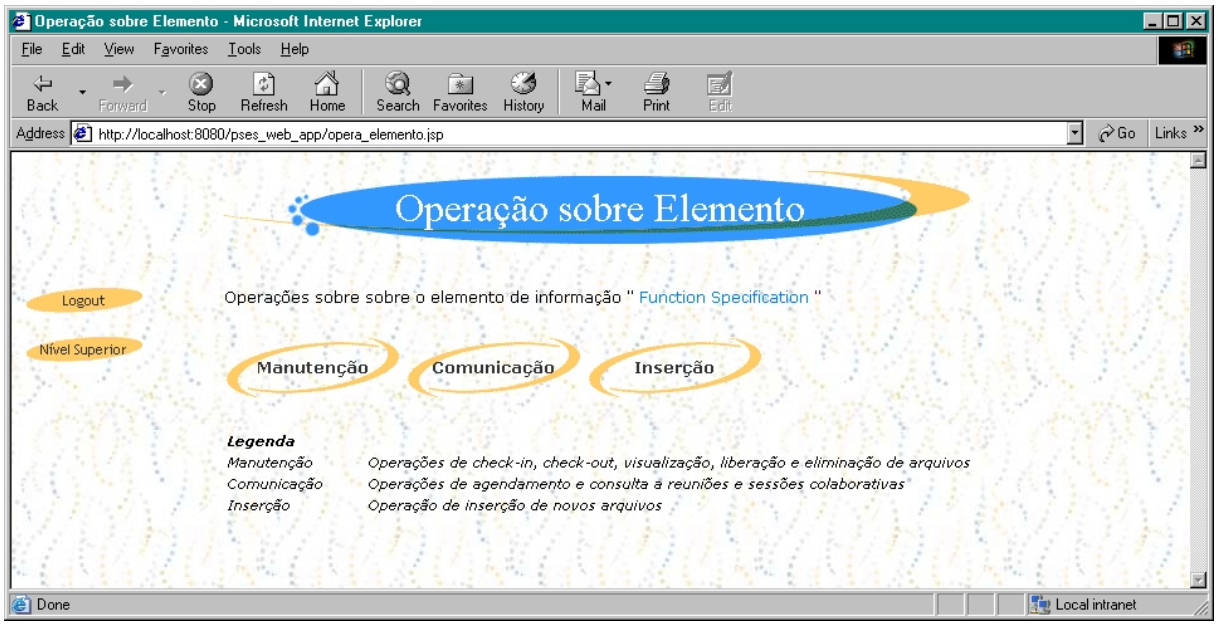


Figura 36 - Operações sobre elementos



A Figura 37 indica as opções para manutenção de arquivos. O *check-in* de um arquivo corresponde ao processo de *upload* de uma versão atualizada de um arquivo constante da base de dados. O *check-out* é o *download* da última versão constante do banco de dados para uma área local de trabalho a ser definida pelo usuário. A visualização é uma operação de *download* apenas do arquivo de visualização vinculado ao arquivo principal, além da visualização imediata no

navegador. A eliminação de um arquivo consiste da eliminação dos registros do mesmo no banco de dados. Finalmente, a liberação é uma operação que permite a confirmação de consistência do arquivo em relação às demais informações de projeto.

Figura 37 - Operações de manutenção sobre arquivos

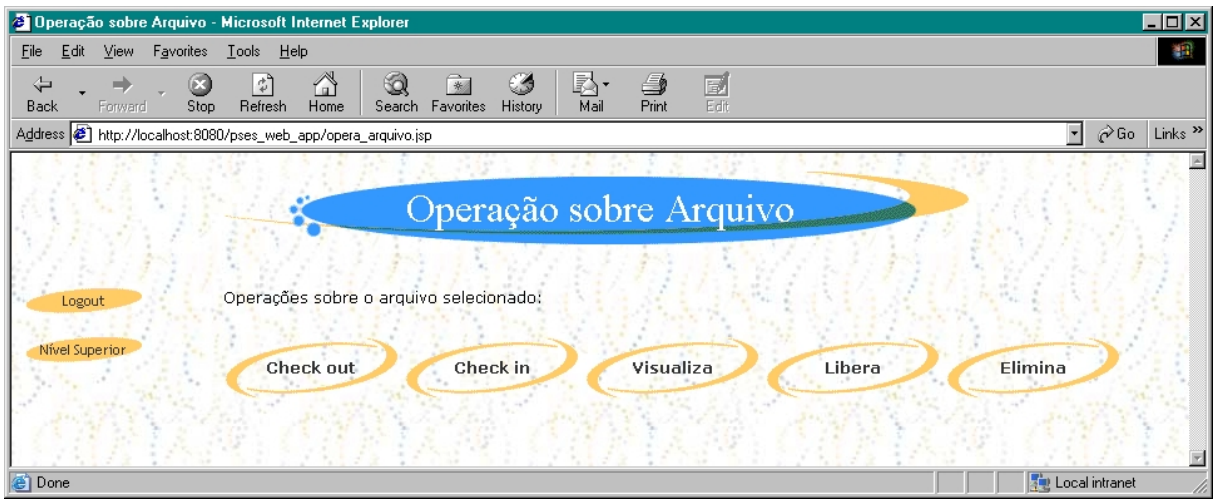
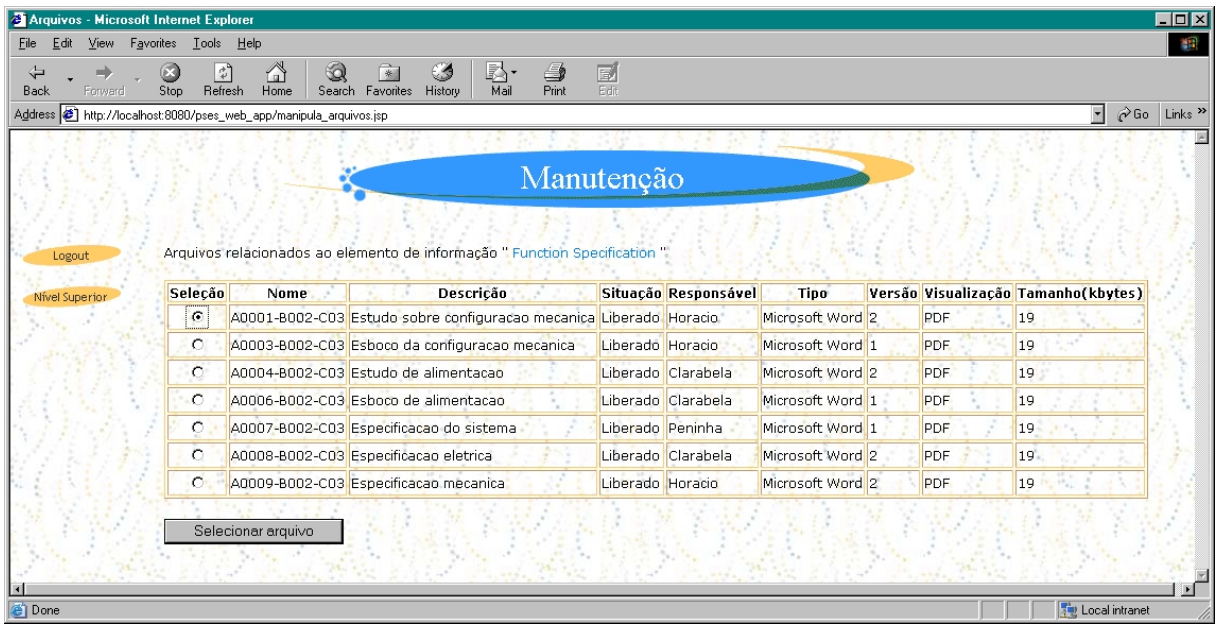


Figura 38 - Apresentação dos arquivos de um elemento de informação para manutenção

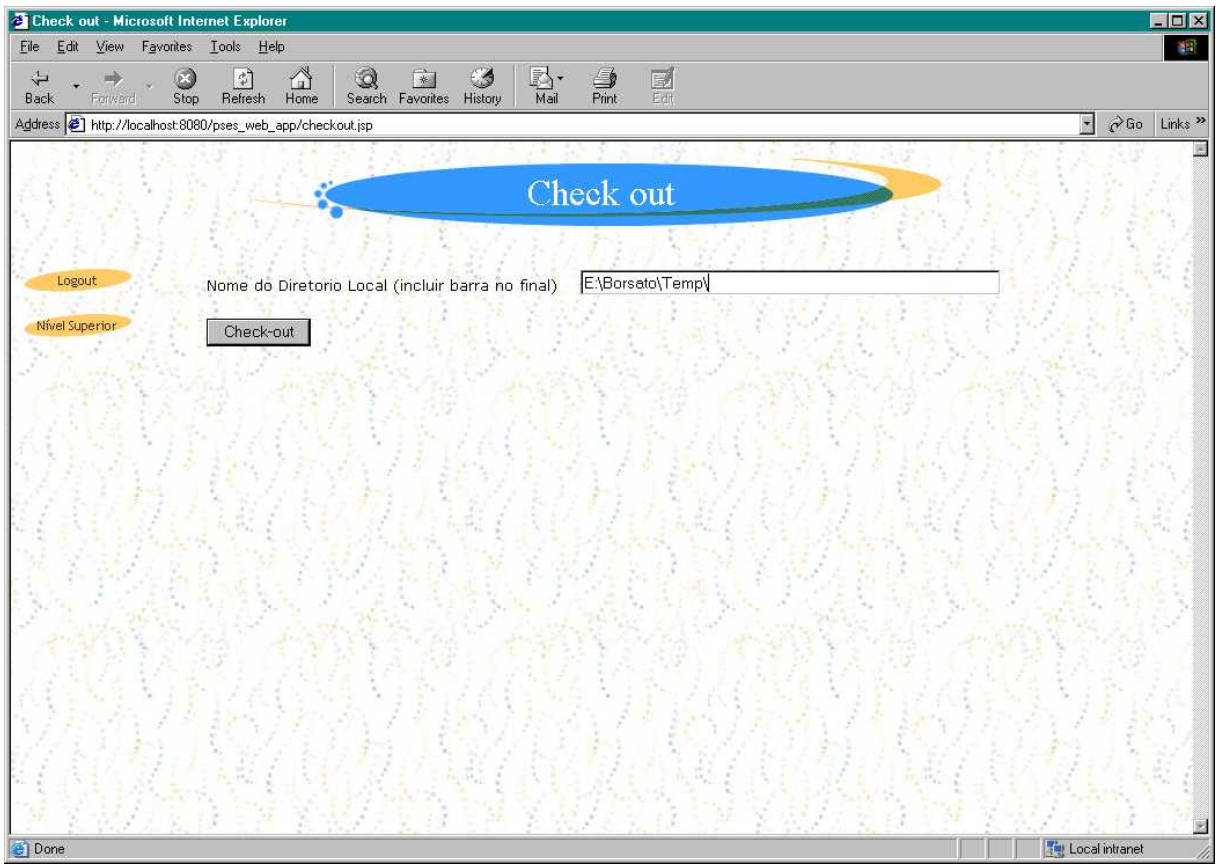


Uma vez definida a operação de manutenção, os arquivos referentes ao

elemento de informação são apresentados, conforme ilustra a Figura 38. Caso ainda não haja arquivos inseridos no elemento, uma tabela vazia é apresentada. As versões dos arquivos constantes da tabela e respectivo tamanho em *kbytes* são as mais recentes.

Um arquivo pode estar enquadrado em três situações. O arquivo **liberado** é aquele que pode ser trazido para a área de trabalho local do colaborador responsável através de uma operação de *check-out*, ilustrada na Figura 39. Por outro lado, o arquivo **bloqueado** é aquele que já foi trazido para a área de trabalho local do colaborador responsável e novas operações de *check-out* não podem ser executadas. Por fim, o arquivo **impactado** é aquele que deve ser verificado pelo colaborador responsável, pois outros elemento de informação dos quais depende o elemento do arquivo foram modificados.

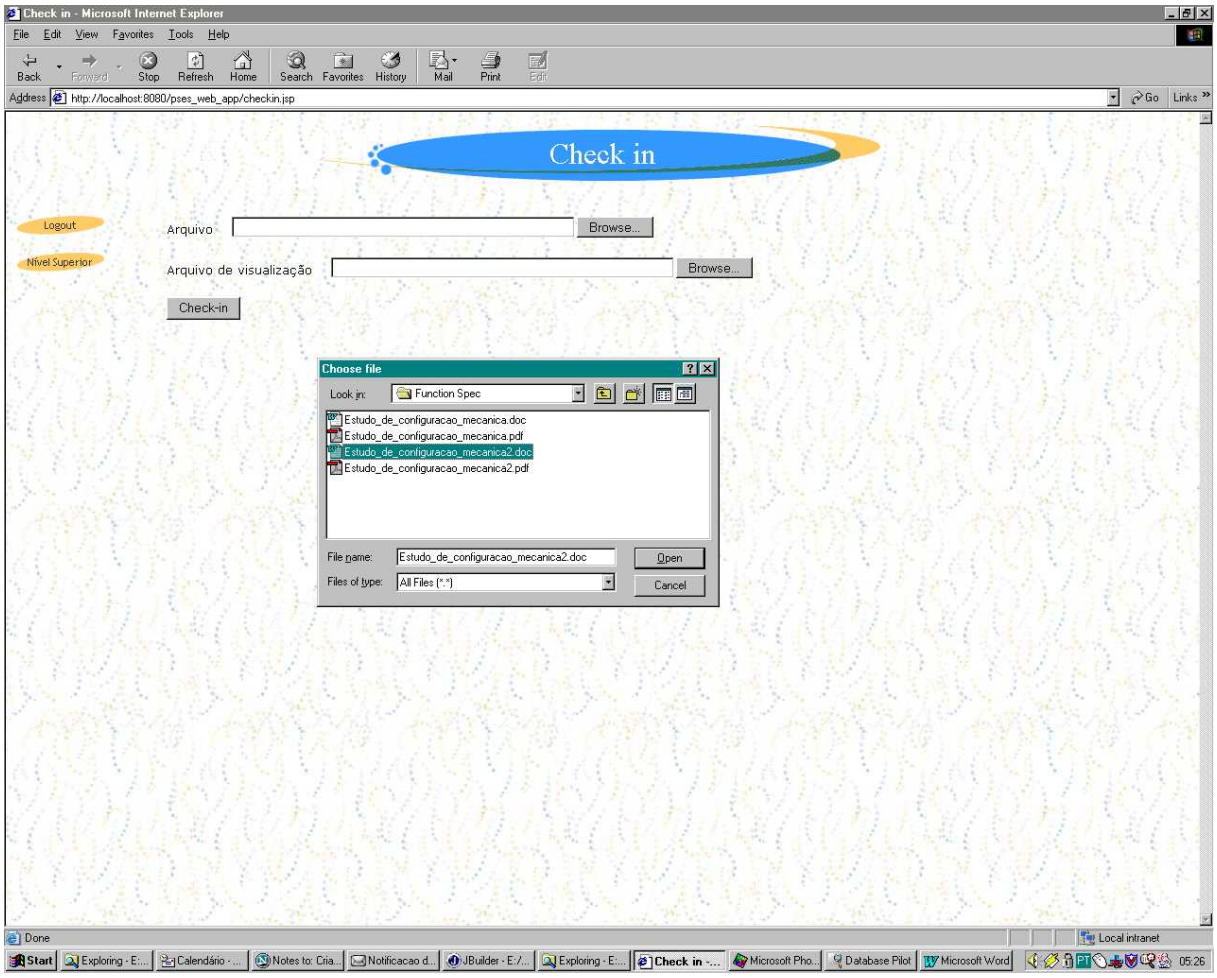
Figura 39 - Operação de *check-out*



Por outro lado, a operação de *check-in* implica no *upload* de dois arquivos: o

fonte e o de **visualização**. A localização dos mesmos na área de trabalho local é feita através de janelas de navegação do sistema operacional local. A Figura 40 indica este processo. Assim que ocorre o *upload* dos arquivos, um novo registro de histórico do arquivo é inserido, contendo a versão e dados de tamanho atualizados.

Figura 40 - Operação de *check-in*



As operações de *check-out*, *check-in*, liberação e eliminação somente poderão ser feitas pelo usuário credenciado, ou seja, aquele que inseriu o arquivo em questão durante a realização de uma tarefa que prevê a manipulação do elemento de informação que o contém. A visualização pode ser feita por qualquer colaborador da equipe de desenvolvimento. Qualquer tentativa desautorizada de operação sobre um arquivo resulta em permissão de acesso negado.

A operação de *check-out* ocorre conforme ilustrado na Figura 39. Uma vez efetuado o *download* para a área indicada, o arquivo é bloqueado na base de dados e ocorre uma confirmação do sistema.

Operações de *check-in* provocam a modificação da situação de arquivos de elementos dependentes para **impactados**. Por exemplo, se um arquivo que contém modelos geométricos 3D do produto são alterados, os arquivos de elementos de informação que dependem do mesmo, como modelos 3D do ferramental a ser utilizado deverão ser verificados. Possivelmente deverão ser atualizados ou, no mínimo, liberados após verificação. A notificação dos responsáveis pelos arquivos impactados é feita por *e-mail*, conforme ilustrado na Figura 41, onde o fato é registrado como evento do projeto.

A operação de visualização também implica num *download* de arquivo, para uma área de trabalho indicada pelo usuário, similarmente à operação de *check-out*. Entretanto, se o *plug-in* correto estiver instalado no navegador do usuário, o arquivo de visualização é aberto diretamente no mesmo, num janela adicional.

Figura 41 - Notificação referente a arquivos impactados

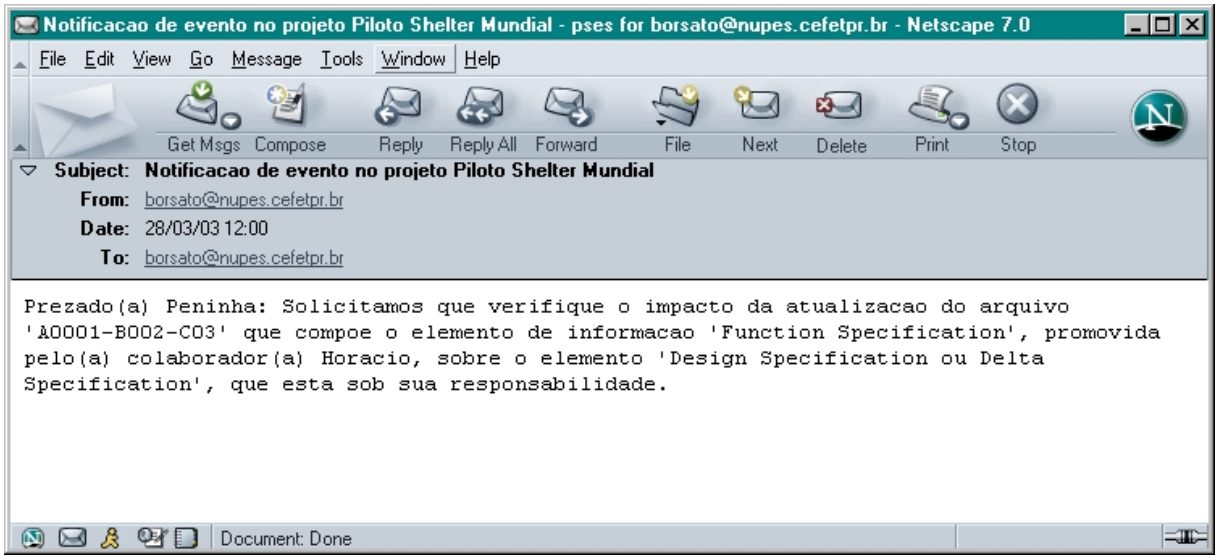


Figura 42 - Notificação referente a arquivos liberados

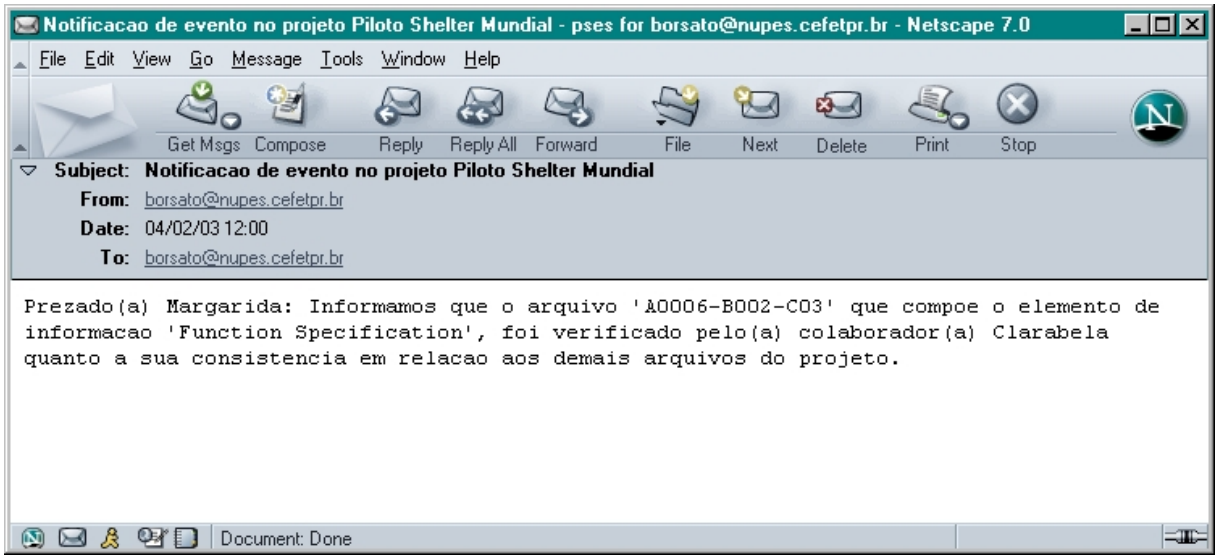
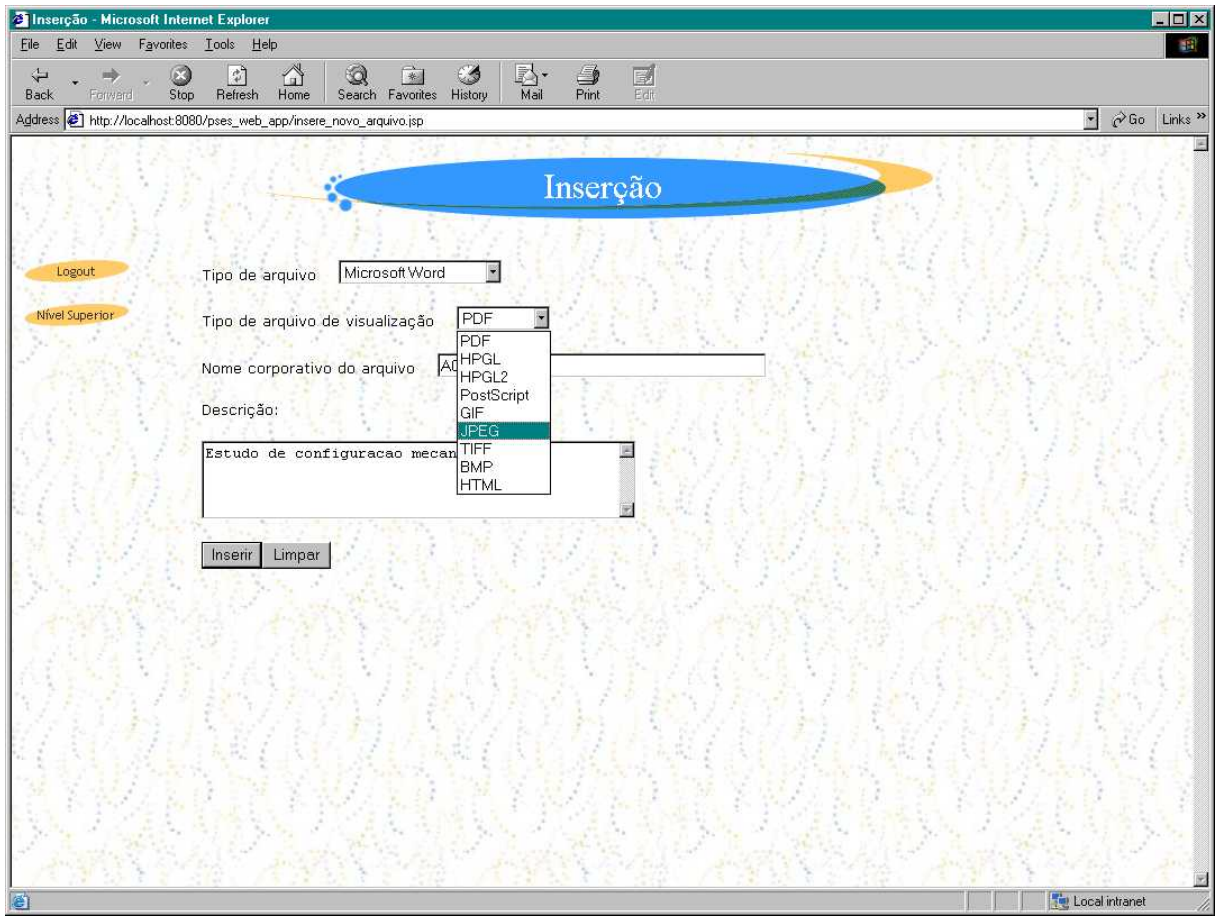


Figura 43 - Inserção de novos arquivos

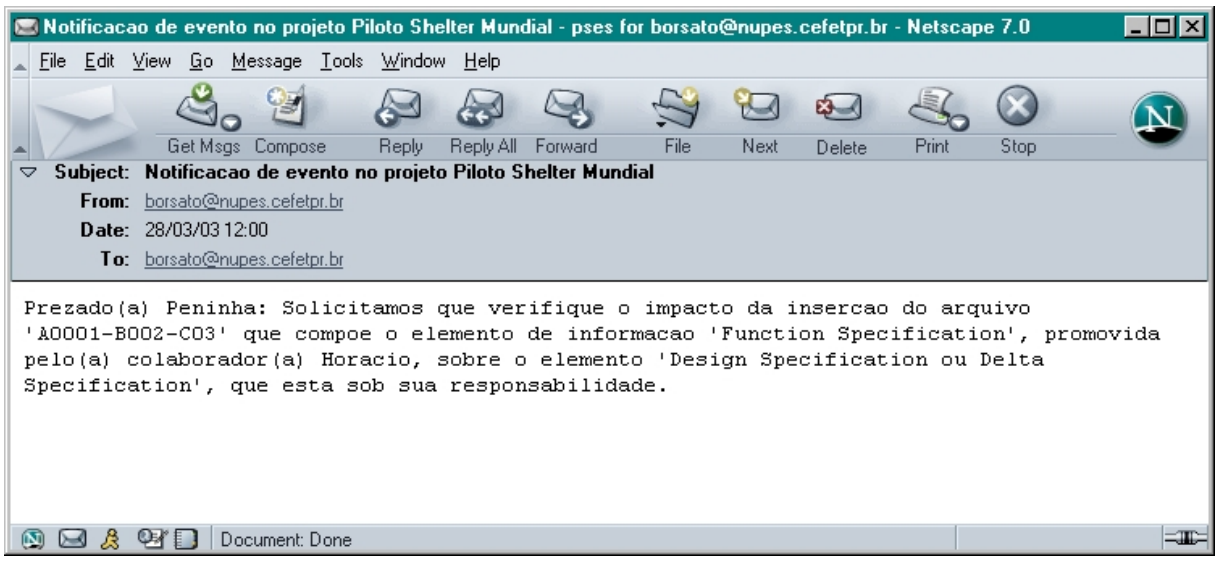


Tal procedimento é realizado através de uma função escrita em JavaScript. Navegadores que não suportem este recurso não apresentarão este comportamento, restando ao usuário abrir manualmente o arquivo de visualização cujo *download* foi executado.

A operação de liberação de um arquivo deve ser feita quando o usuário responsável por um elemento é notificado. Isto pode ser feito através do recurso de liberação de arquivo dentro das opções de manutenção. A liberação de um arquivo também provoca notificação dos colaboradores afetados por *e-mail*, conforme ilustrado na Figura 42.

Assim como a operação de *check-in* de arquivos, a inserção de novos arquivos requer que o usuário identifique uma série de propriedades e efetue o *upload* a partir de uma área local de trabalho. A Figura 43 traz o processo de inserção de novos arquivos e a Figura 44 traz a notificação gerada aos usuários impactados.

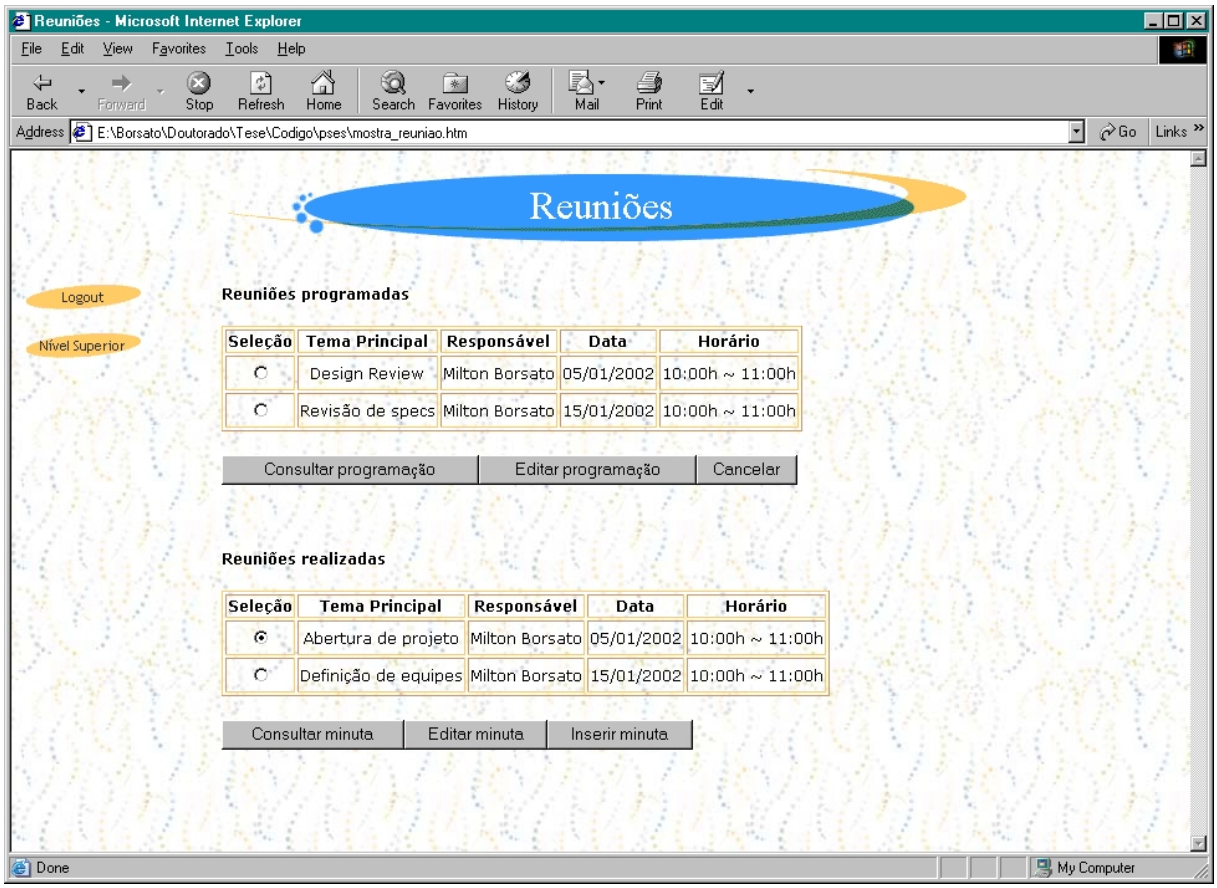
Figura 44 - Notificação referente a arquivos novos inseridos



No sistema PSES, a opção de operação sobre elementos de informação é a **comunicação**, conforme ilustrado na Figura 36. Esta opção consiste em facilidades para o agendamento, registro e consulta de reuniões convencionais (*off- line*) e sessões de trabalho colaborativo via sistema (*on-line*). As reuniões convencionais podem ser

programadas em sua pauta, data, horário e participantes. A Figura 45 apresenta as opções referentes a reuniões convencionais. A programação de uma reunião consiste em especificar uma pauta de assuntos, participantes convidados, data e horário de realização. Os participantes são notificados por *e-mail*. Uma vez realizada a reunião, uma minuta pode ser inserida ou mesmo editada. A programação e inserção ou edição de minutas gera eventos e notificações. As minutas de reuniões já realizadas podem ser consultadas a qualquer momento. A Figura 46 mostra como uma minuta pode ser inserida.

Figura 45 - Apresentação de reuniões



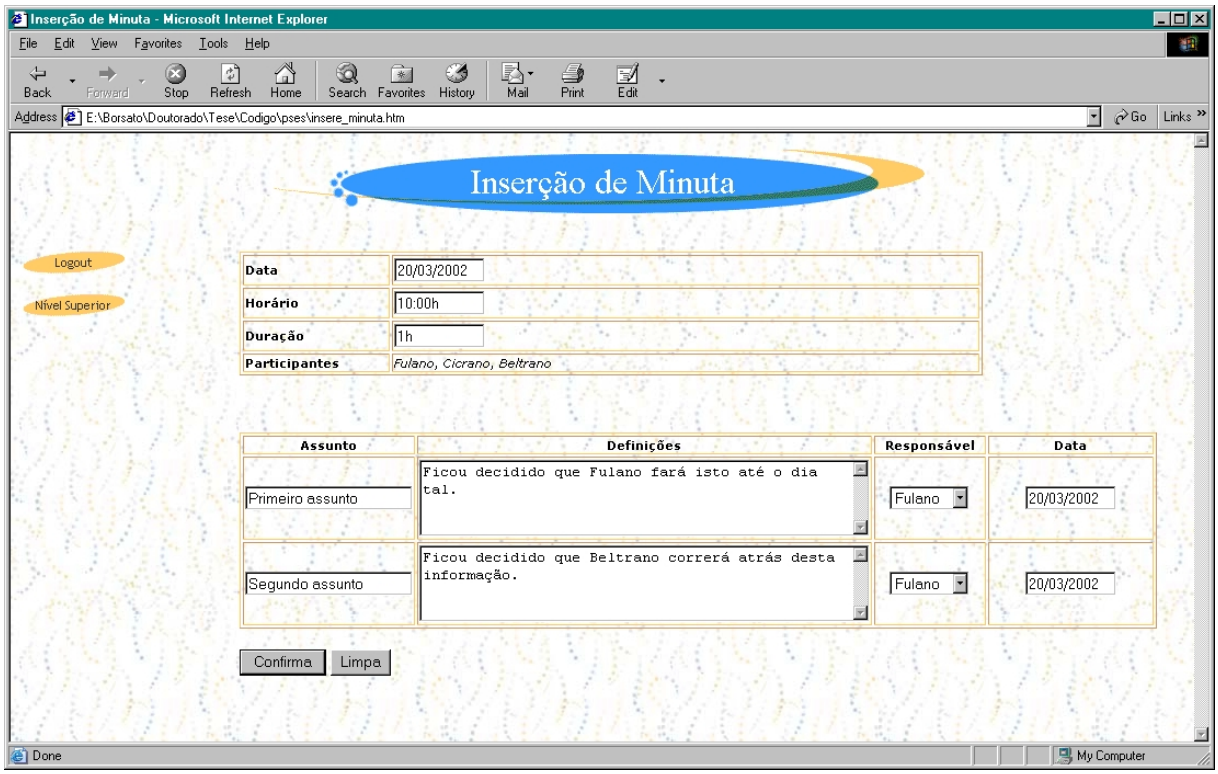
As sessões colaborativas são estabelecidas para que, mesmo que remotamente, os participantes apresentem documentos de visualização aos demais e elaborem comentários e sugestões, tanto de forma escrita (*chat*) como de forma gráfica (*markup*). Ao ingressar numa sessão colaborativa, o colaborador pode postar as figuras

que desejar e escolher uma cor para suas anotações.

Os comentários e anotações são compartilhados simultaneamente pelos demais participantes. As intervenções ficam registradas através de um arquivo de *log* que pode ser consultado posteriormente. A Figura 47 apresenta um exemplo de sessão colaborativa.

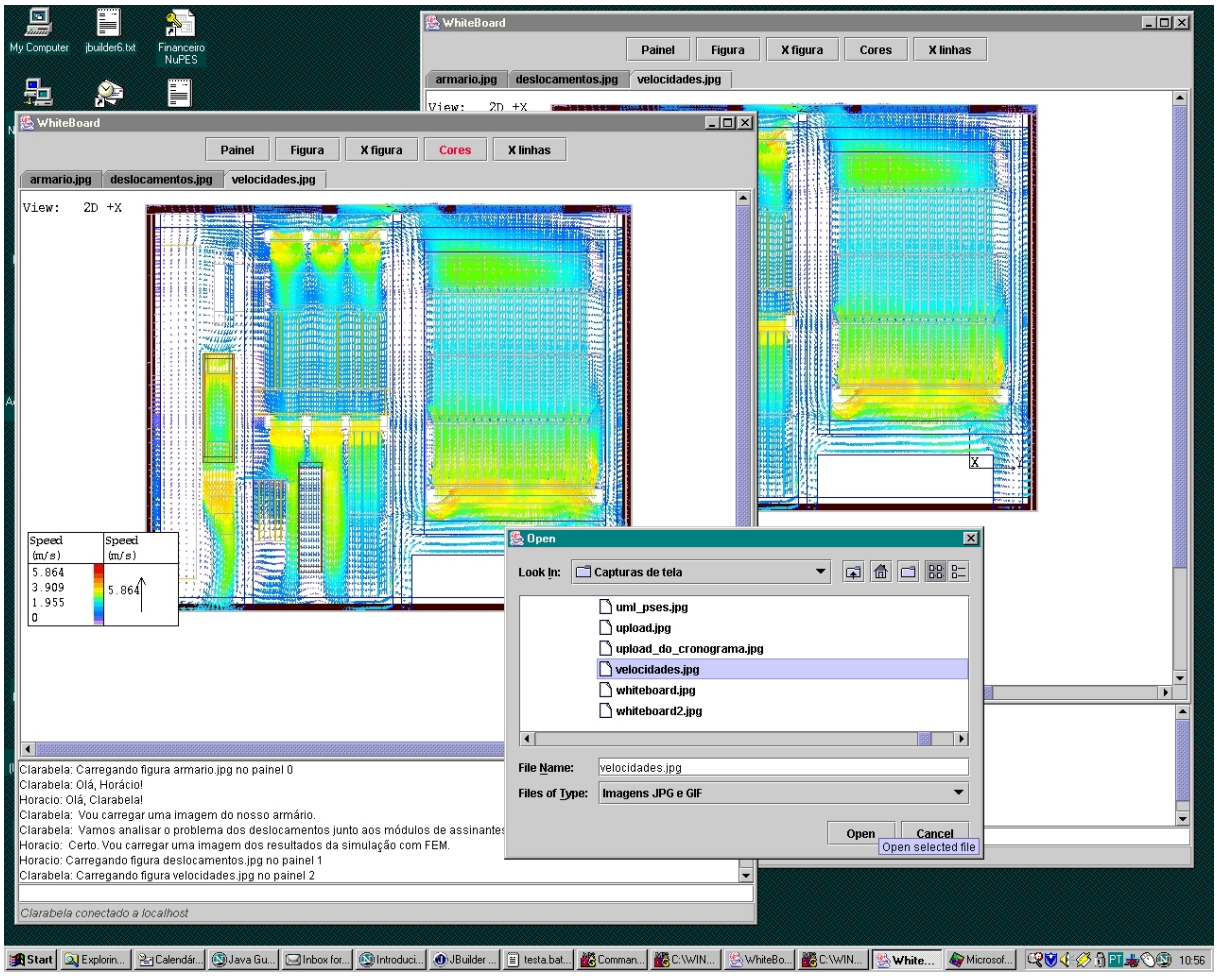
A sessão colaborativa é implementada através de um *applet* Java que opera na máquina cliente. Neste caso, o *applet* funciona independentemente do navegador. As anotações (*markups*) são convertidos em informações vetoriais para minimizar a quantidade de informações que deve trafegar na rede.

Figura 46 - Operação de inserção de minuta de reunião



Tanto as reuniões quanto as sessões colaborativas são vinculadas a elementos de informação, de forma que estas informações podem ser utilizadas posteriormente para rastrear uma decisão tomada num determinado momento do processo de desenvolvimento.

Figura 47 - Exemplo de sessão colaborativa



4.4 MODO DE GESTÃO

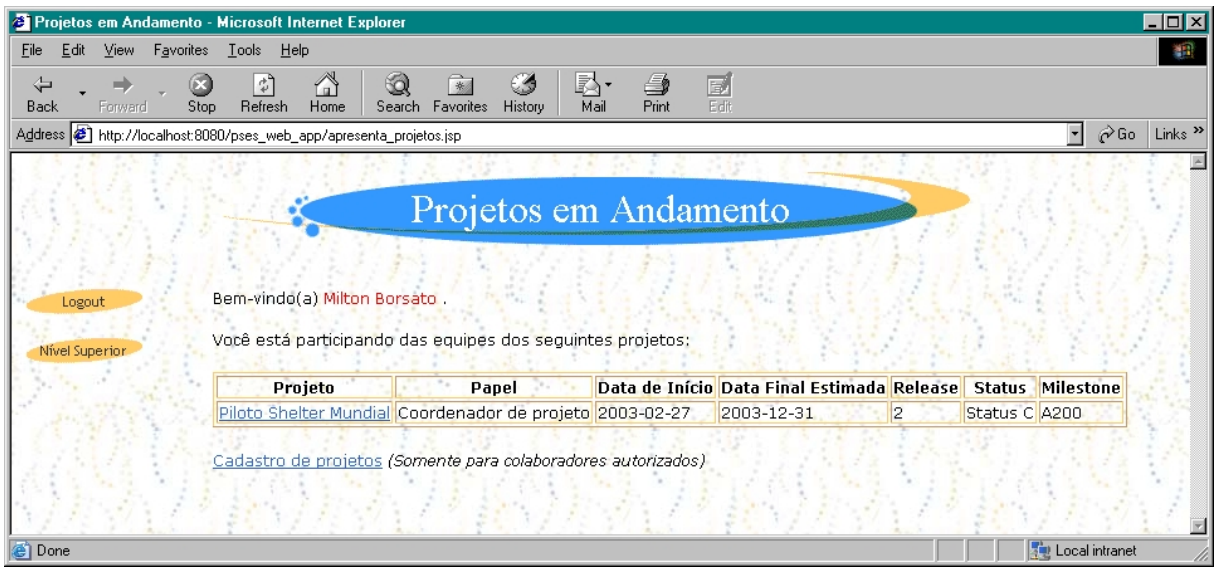
O modo de gestão é o último dos três casos de uso para o sistema, conforme ilustra a Figura 12. No sistema PSES, os projetos devem ser cadastrados por um usuário do grupo **Project managers**. O gerente de projeto pode incluir, reconfigurar ou eliminar um projeto. Pode ainda executar as operações disponíveis aos usuários comuns, desde que observadas as responsabilidades por tarefas no projeto. Há, entretanto, operações exclusivas ao gerente, tais como a manipulação de elementos mestres do projeto.

A função de gerente ou coordenador do projeto específico que está sendo

cadastrado poderá ser posteriormente atribuída a outro colaborador do mesmo grupo, porém sempre deverá haver um único colaborador nesta função por projeto.

Ao efetuar o *logon* no sistema, caso o usuário em questão pertença ao grupo **Project managers**, a opção de cadastramento de projetos é habilitada durante a apresentação de projetos. Esta situação é apresentada na Figura 48.

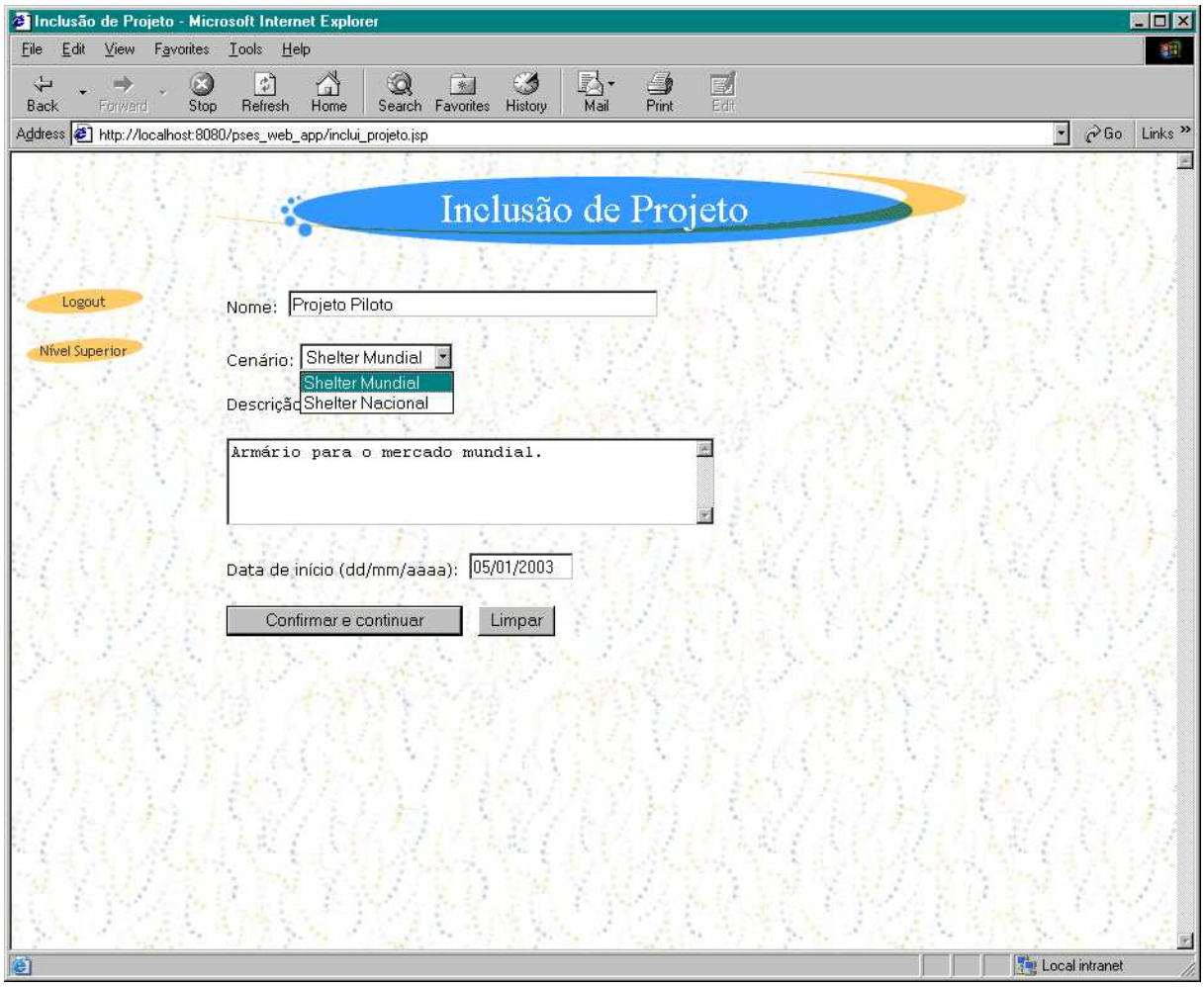
Figura 48 - Visão do gerente para a apresentação de projetos



A inclusão de um projeto se dá em etapas. Primeiramente, o gerente deverá incluir dados gerais do projeto, como nome e descrição, além de escolher o cenário segundo o qual o projeto deverá ocorrer, conforme apresentado na Figura 49. A data de início do projeto é apenas uma referência inicial. Esta data é sincronizada com o cronograma mestre do projeto ao final do processo de inclusão do projeto.

O passo seguinte para a inclusão de um novo projeto no sistema PSES é a constituição da equipe de colaboradores e atribuição de papéis aos membros. Isto é realizado através de escolha por menus do tipo *pull-down*, nos quais as opções são montadas dinamicamente segundo os itens já cadastrados no banco de dados pelo administrador. A Figura 50 apresenta a operação de criação da equipe de colaboradores de um projeto. A cada vinculação segue-se uma tela de confirmação na qual aparecem os membros já designados, conforme apresenta a Figura 51.

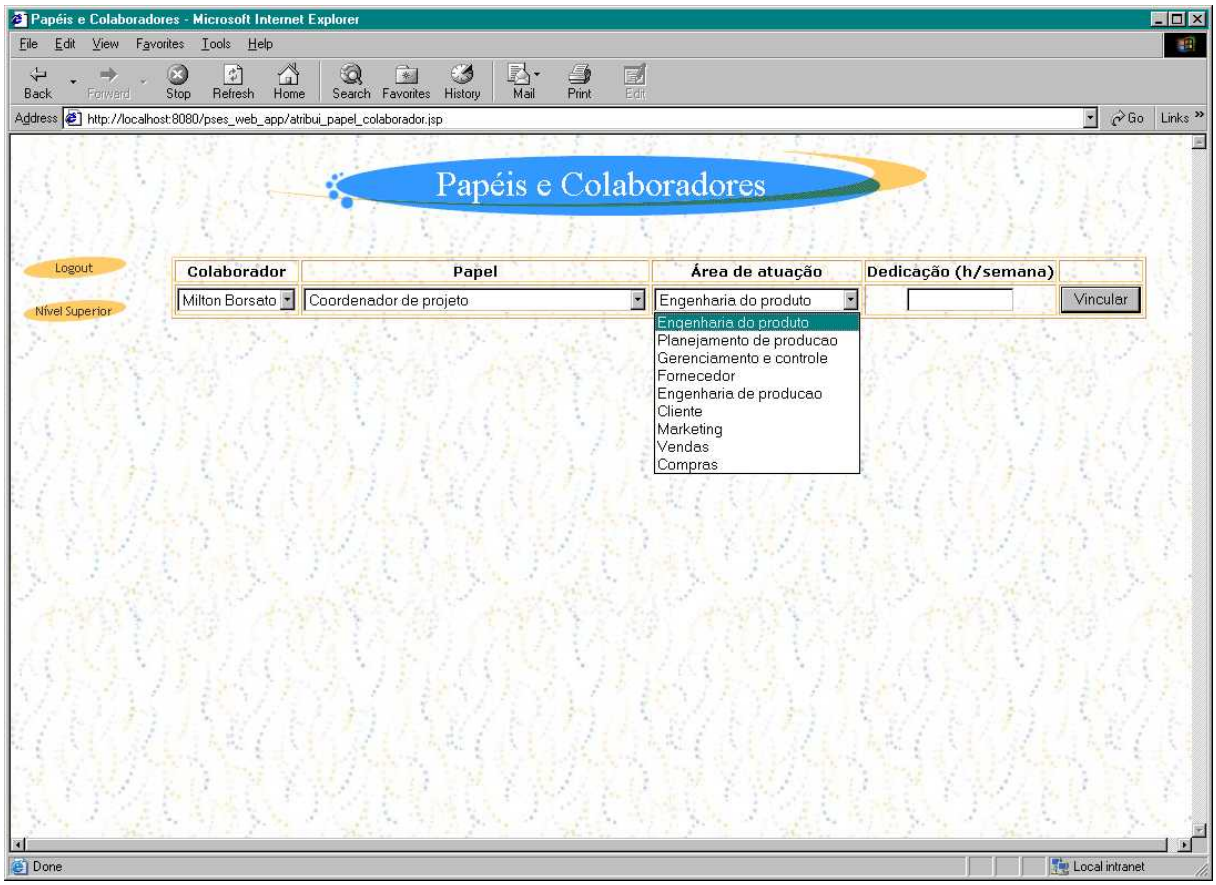
Figura 49 - Inclusão de projetos



Em seguida, o gerente de projeto atribui responsabilidades a cada membro da equipe. Isto é realizado novamente através de menus onde aparecem os colaboradores já escolhidos e as tarefas do cenário segundo o qual o projeto acontece. Somente tarefas ainda sem atribuição de responsável constam do menu *pull-down*. Esta operação pode ser observada na Figura 52.

É possível atribuir várias tarefas a um mesmo colaborador, porém uma mesma tarefa não pode ser atribuída a mais de um membro da equipe. Portanto, o modelo de processos deve ter seu detalhamento feito de forma que, quando se chega a um responsável por cada uma das tarefas, o mesmo estará com a profundidade adequada (ver 3.4.1).

Figura 50 - Escolha dos membros da equipe

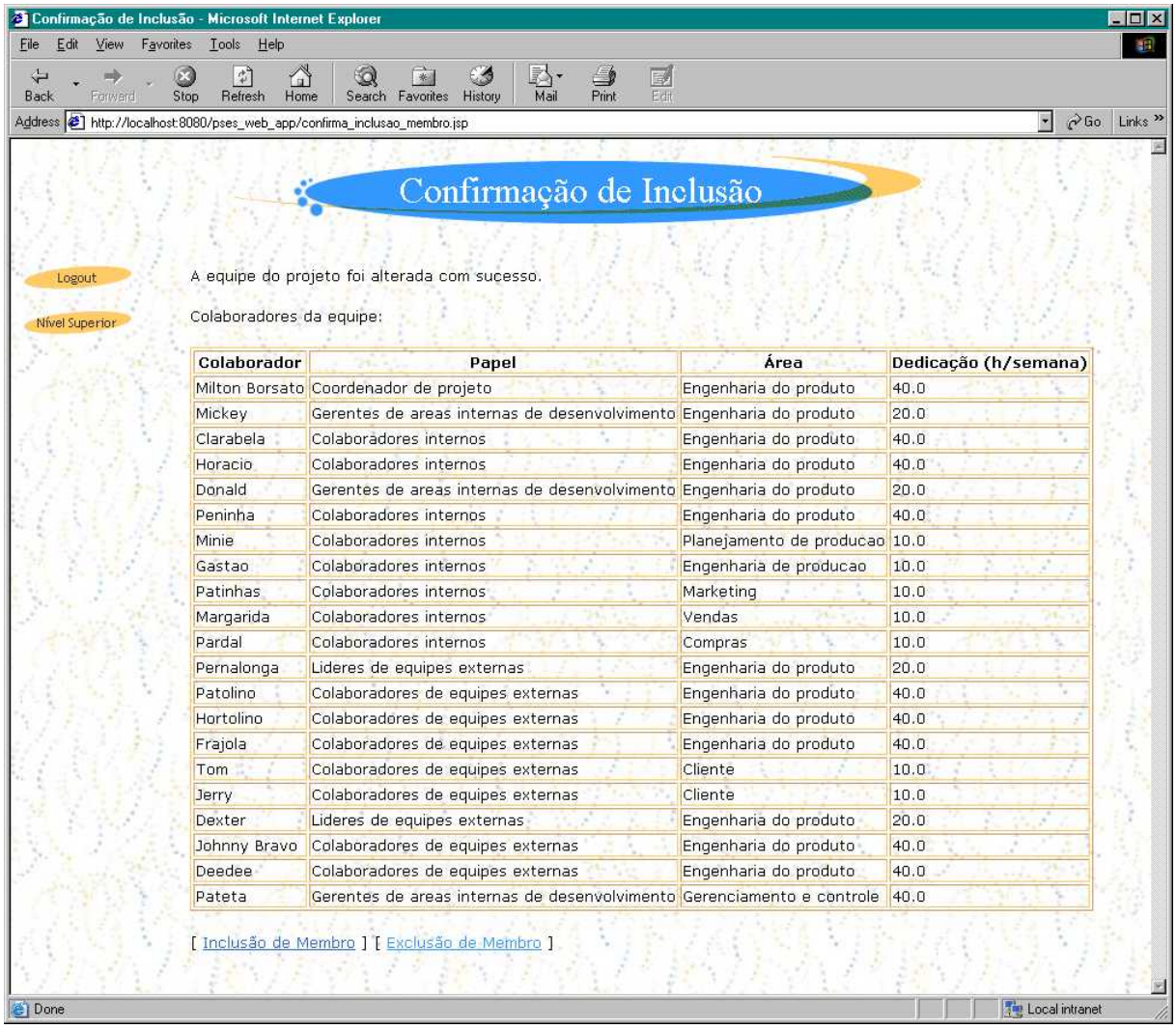


Na seqüência, o gerente de projeto vincula cada ramo da árvore de decomposição do produto a um repositório. Neste momento é que se define de que forma os dados de projeto serão fisicamente distribuídos. Esta vinculação deve ser feita com base pelo menos na distribuição geográfica e tipos de acesso a *Internet* disponível em cada localidade de desenvolvimento, a fim de reduzir eventuais transtornos com desempenho da rede (MANSFIELD et al., 1999). A Figura 53 traz esta operação.

Como último passo na criação de um projeto, o gerente deve promover a atualização de datas e durações das tarefas através do *download* do cronograma padrão do cenário, atualização do mesmo através da ferramenta de gestão de projetos (*eg.* Microsoft Project 2002) e *upload* do arquivo atualizado em formato XML. A Figura 54 apresenta a tela que traz orientações para atualização de datas e durações de tarefas

do cronograma mestre.

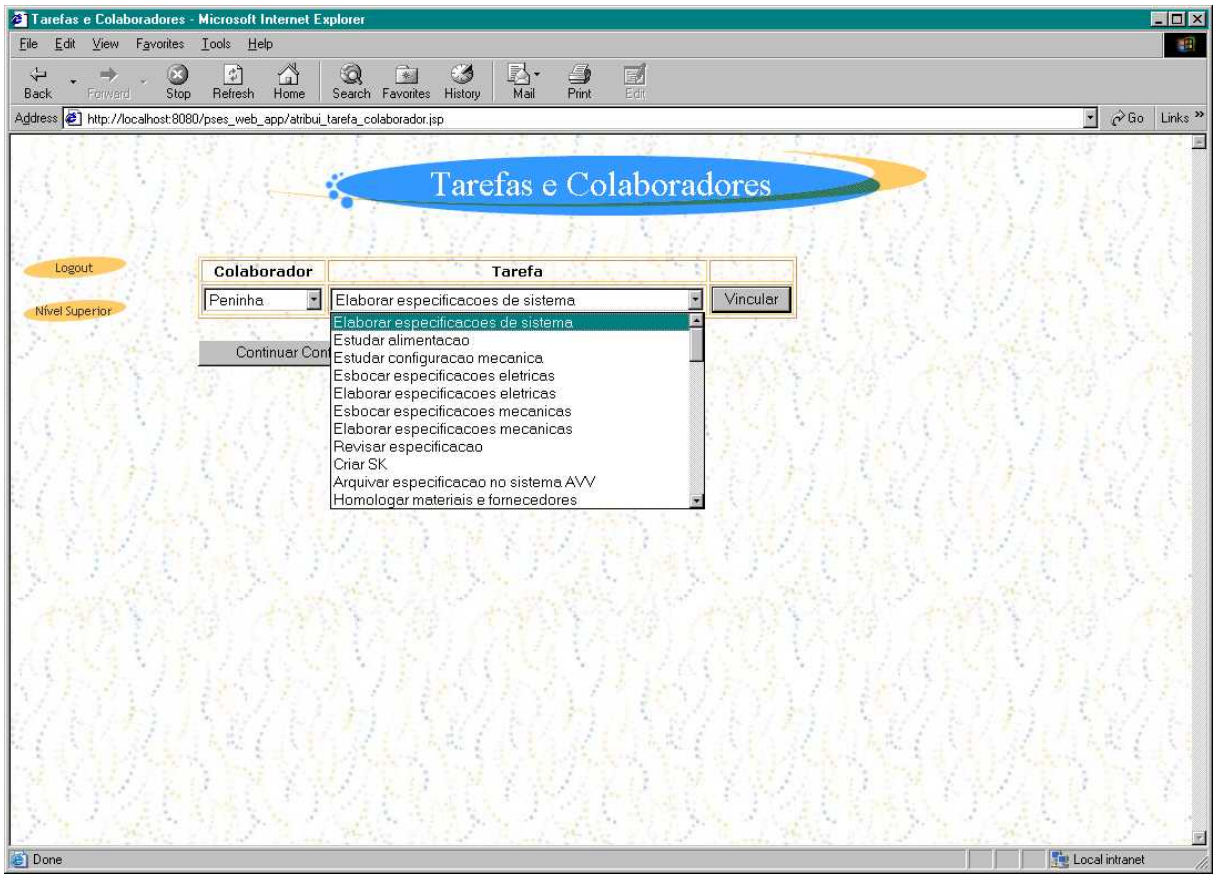
Figura 51 - Confirmação de membros da equipe já designados



Neste momento o sistema promove a atualização das datas de início e fim de cada tarefa através de um mecanismo de *parsing*[19] do arquivo XML e comparação entre nomes de tarefas no sistema e no cronograma, para que o sistema fique sincronizado com o cronograma mestre.

[19] *Parsing* é um processo de separação de campos de um arquivo através de uma seqüência de comparações lógicas.

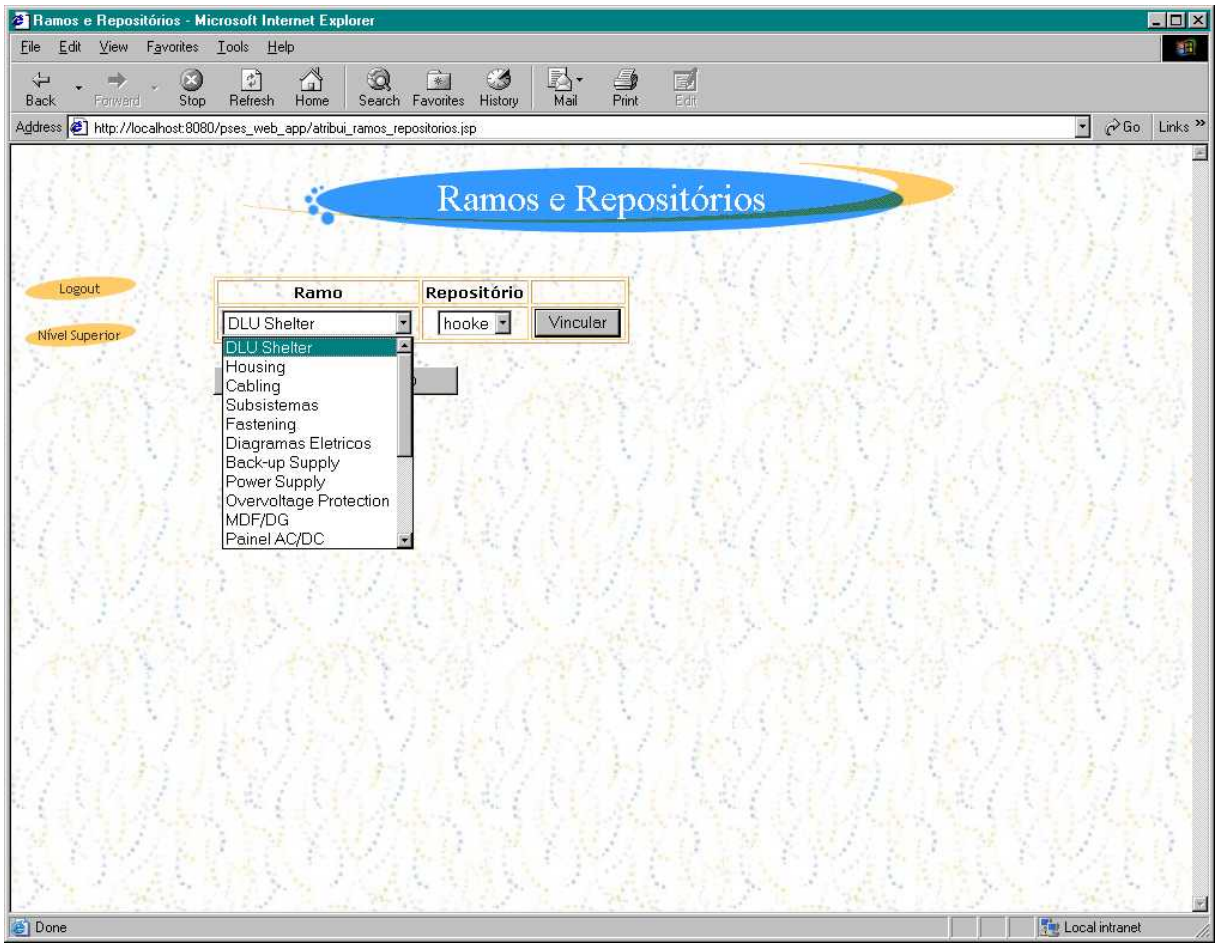
Figura 52 - Atribuição de responsabilidades aos membros da equipe



Por este motivo os nomes de tarefas não possuem acentuação no cronograma mestre, já que acentos não são tratados nos campos de texto do PostgreSQL. Novas sincronizações ocorrerão através de operações de controle executadas pelo gerente de projeto na medida em que forem necessárias. A Figura 55 ilustra a operação de *upload* do cronograma mestre a partir de um diretório local do posto de trabalho.

As configurações inicialmente adotadas, como escolha dos membros da equipe, atribuições de responsabilidades por tarefas, entre outras, podem ser revistas posteriormente, através da reconfiguração de projetos. As opções disponíveis ao gerente de projetos durante a inclusão de projetos em suas diversas etapas dependem diretamente daquelas que o administrador tenha cadastrado, tais como colaboradores e repositórios disponíveis.

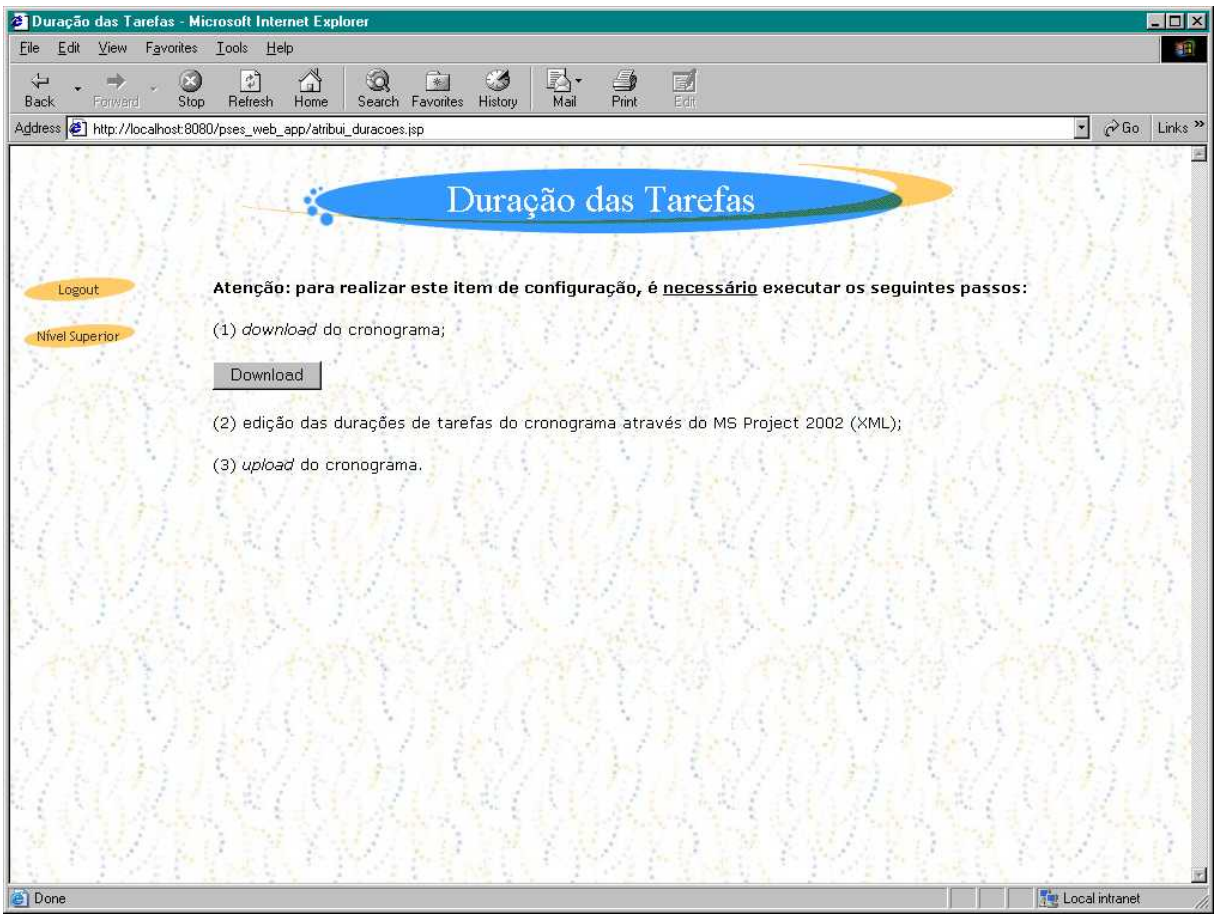
Figura 53 - Vinculação de ramos a repositórios



A reconfiguração pode ser efetuada pelo respectivo gerente, após a criação do projeto. Neste caso, podem ser modificados os membros da equipe de desenvolvimento e atribuições de tarefas. As vinculações devem ser removidas primeiro para que as tarefas sejam reapresentadas para vinculação a membros da equipe.

Durante a realização de um projeto, há determinados elementos que não estão vinculados a uma tarefa específica. São os **elementos-mestres**. Estes elementos abrigam arquivos que dizem respeito ao projeto como um todo, tais como arquivos de cronograma, requisitos do cliente e outros. O acesso a tais elementos pode ser feito através do *link* gestão, que aparece na Figura 27. A Figura 56 apresenta as opções de gestão: controle, elementos, métricas e *release*.

Figura 54 - Orientações para *upload* do cronograma mestre



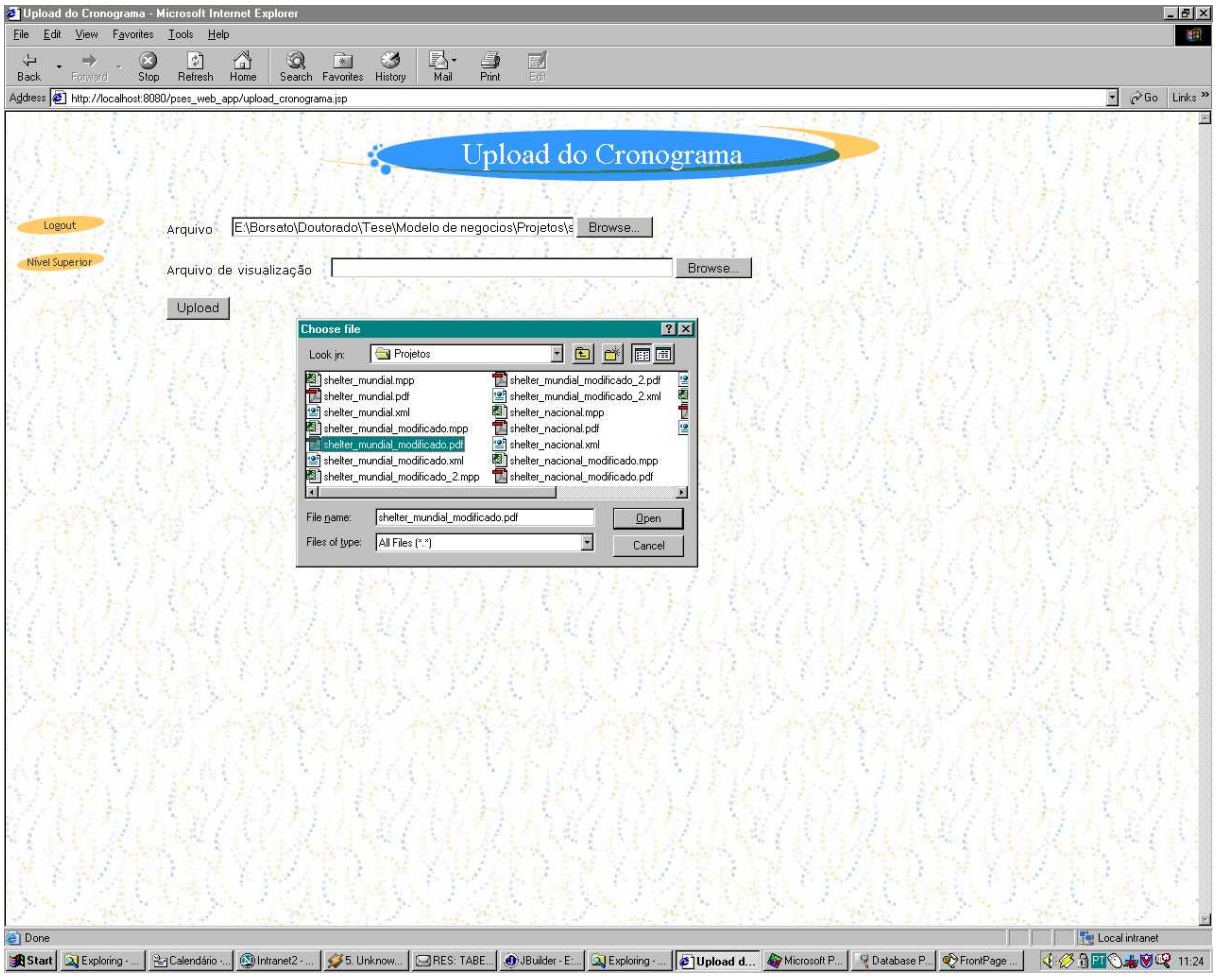
Na opção **elementos**, podem ser acessados os elementos-mestres do projeto, para visualização, no caso do usuário comum e para modificação, no caso do gerente do projeto. Os procedimentos para *check-in*, *check-out*, visualização e eliminação de arquivos são idênticos aos já descritos.

Na opção **controle**, existe a possibilidade de sincronização da base de dados do sistema como cronograma mestre e vice-versa. Basicamente, o gerente de projetos deve ter atualizado o cronograma mestre através do processo habitual de *check-out* e *check-in*. A Figura 57 mostra o processo de seleção das operações de controle.

A operação de sincronização da base de dados consiste num processo de *upload* do arquivo XML referente à última versão do cronograma mestre, desde o repositório central (tronco) para uma área temporária que fica na árvore de diretórios

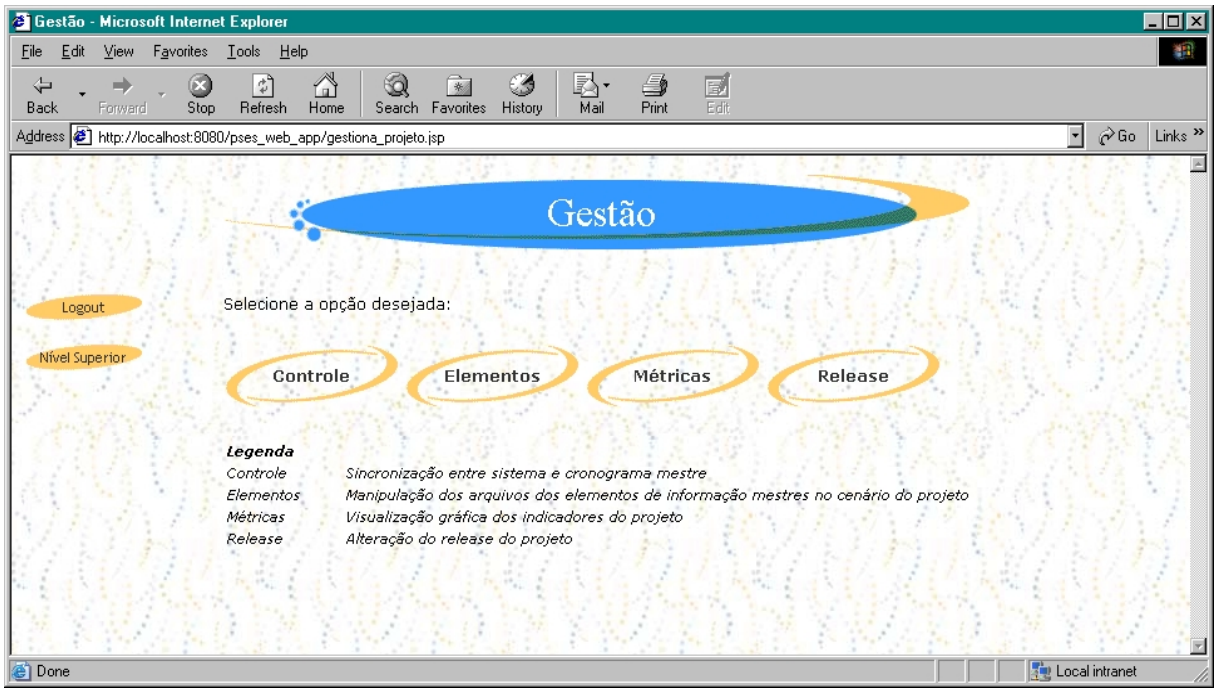
do contêiner JSP (ver 3.4.6) e *parsing* do mesmo. Durante esta etapa, são extraídas as informações de datas e durações das tarefas. Novos históricos das tarefas são então inseridos contendo as informações atualizadas.

Figura 55 - *Upload* do cronograma mestre atualizado



A operação de sincronização de cronograma mestre com base nas informações do sistema consiste no processo reverso. Porém, neste caso, apenas as informações de percentuais concluídos das tarefas são repassadas ao arquivo XML, cujo *download* é feito para uma área local de trabalho do gerente de projeto. Cabe ao gerente de projeto, em sua ferramenta de gestão de projetos (*eg.* **Microsoft Project**), recalcular as datas e durações conforme desejar.

Figura 56 - Opções de gestão



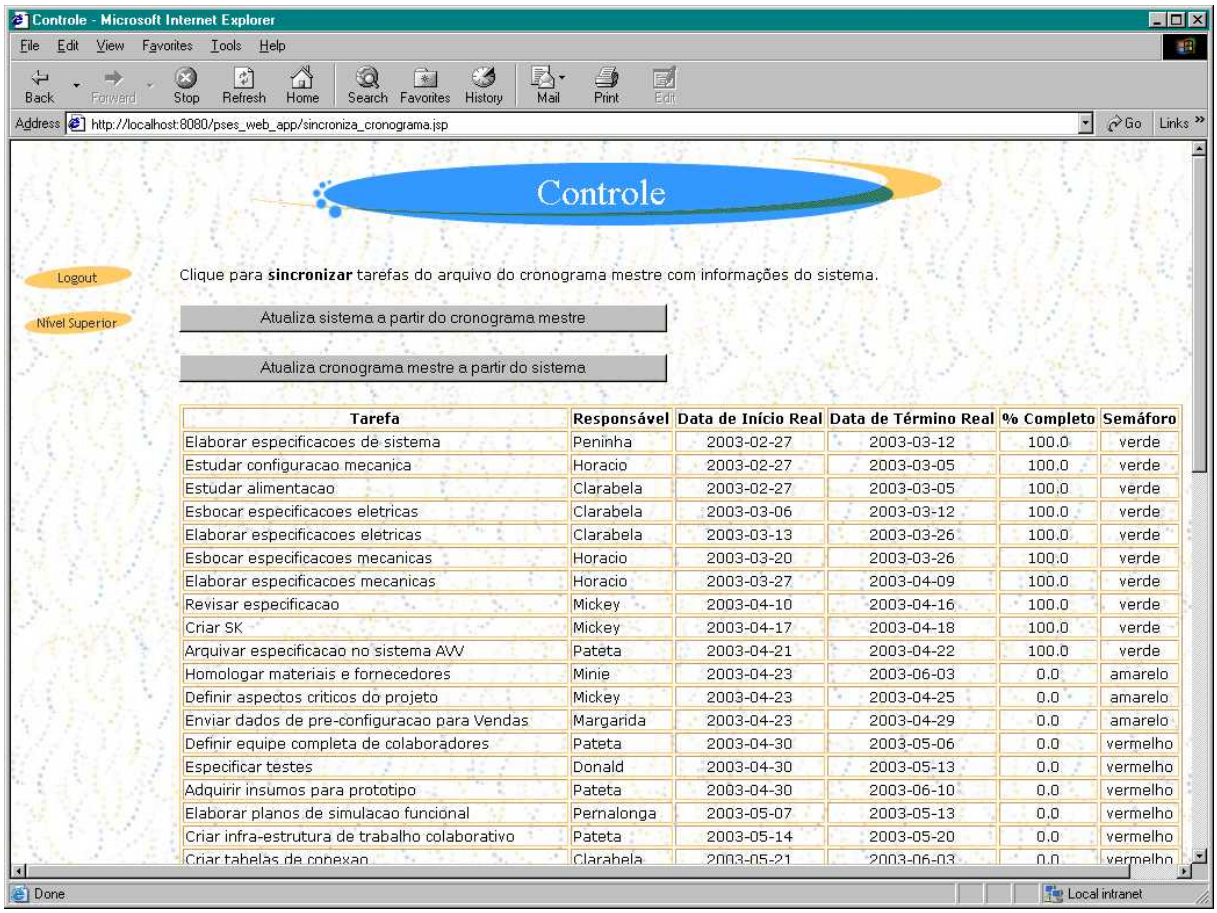
É importante ressaltar que o sistema PSES não pretende substituir ferramentas de gestão de projetos, e sim permitir que sua funcionalidade seja aproveitada num sistema de suporte mais ambicioso para o desenvolvimento de produtos, segundo os preceitos da ES. Após esta sincronização, o gerente de projeto pode efetuar a atualização do cronograma mestre no sistema, sincronizar com a base de dados e, assim, fechar o ciclo.

Outra possibilidade de gestão é a sincronização do modelo de informação através da progressão de *releases*. O *release* deve ser interpretado, no caso do sistema PSES, como uma forma de informar o colaborador da equipe que num determinado momento os dados do projeto estão consistentes entre si, e, portanto, o modelo de informação do produto pode ser **promovido** a um novo *release*.

A condição para que o gerente de projeto promova o avanço de *release* do projeto é de que não haja arquivos em situação de **impactado** ou **bloqueado**. O sistema não faz isto de forma automática porque entende que seja o gerente de protejo quem deva executar esta tarefa. O sistema atua apenas no sentido de informar quais arquivos estão impedindo um avanço de *release*. Uma vez atualizado o *release*, o

sistema gera um evento, atualiza a base de dados e informa os colaboradores do fato. A Figura 58 mostra a situação na qual o *release* pode ser atualizado. Caso haja algum arquivo inconsistente, a opção de atualização não é apresentada e uma nota explicativa indica que a operação não é possível.

Figura 57 - Opções de controle

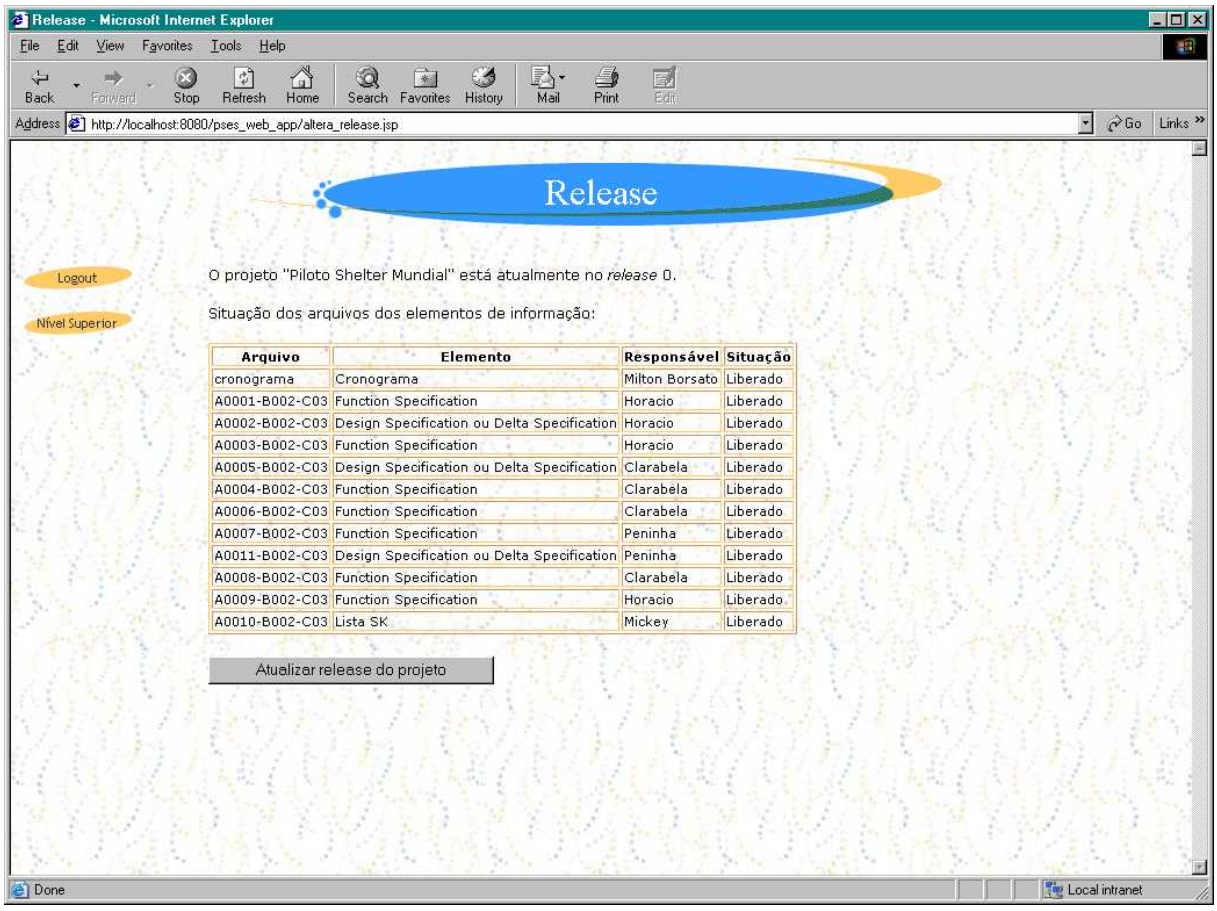


Dentre as opções de gestão, existe ainda o acesso ao sistema de métricas proposto neste trabalho. Para efeito de codificação, foram escolhidas as métricas IIX, IPT, NAE, NSD e IDC, onde IIX é o índice de integração de áreas, IPT é o índice de paralelismo entre tarefas, NAE é o número de alterações de especificação, NSD é o número de sincronizações da base de dados e IDC é o índice de desvio de cronograma.

Em relação ao previsto no item 3.4.3, foram unificadas as métricas IIM e IIF numa métrica referenciada como IIX, que avalia não só a integração de manufatura e

fornecedores com o restante da equipe, mas também as demais áreas e as compara através de um único diagrama. Isto se deve à facilidade de composição de um diagrama de várias linhas, onde cada uma representa uma área. Além desta modificação, a métrica IPT substituiu a métrica IPF proposta, em função da plataforma ser destinada a vários cenários, nos quais nem sempre as fases de projeto coincidem.

Figura 58 - Atualização de *release*



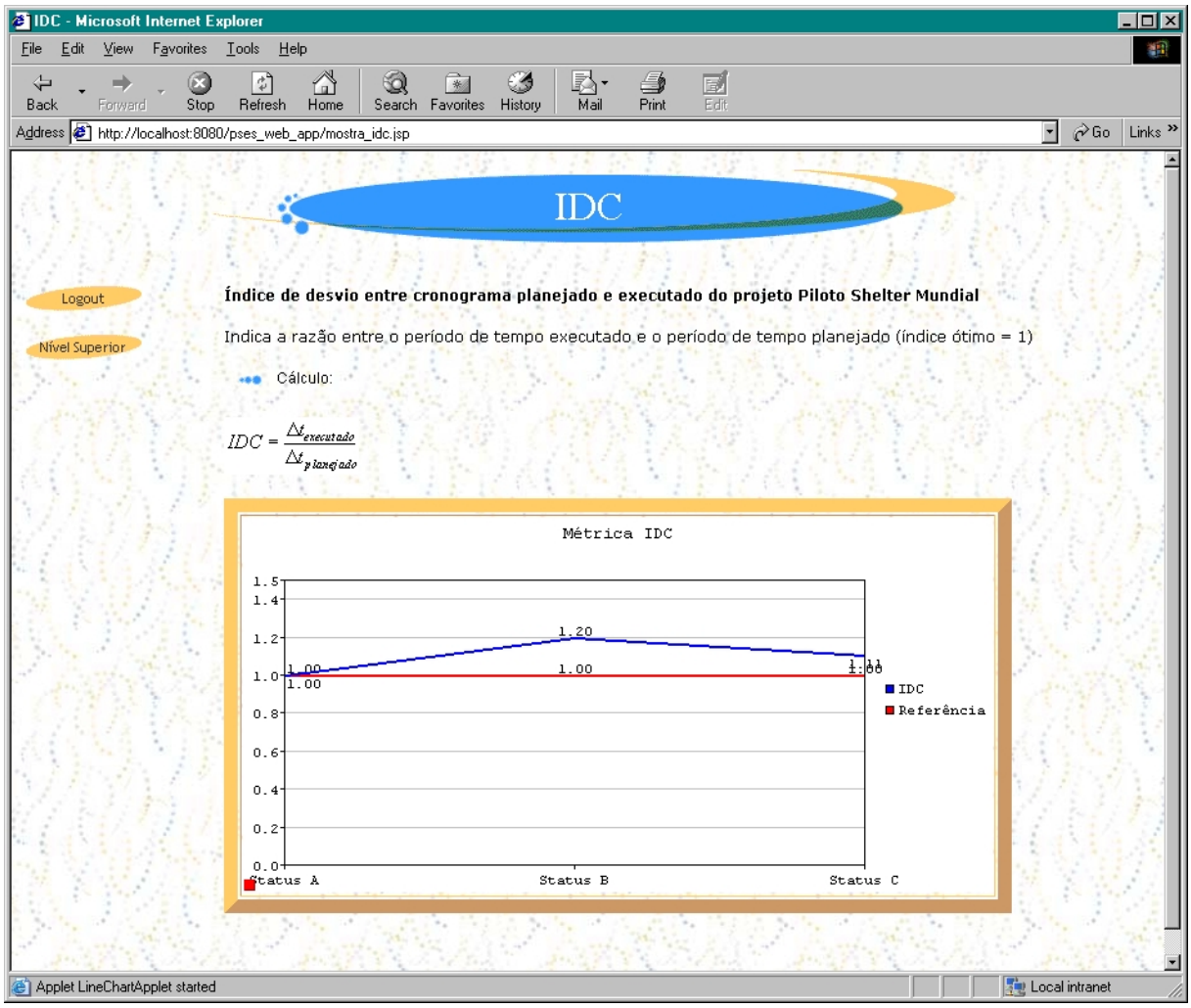
A representação gráfica das métricas é feita sob demanda. Ocorre a consulta à base de dados e as informações necessárias para a construção dos gráficos são reunidas para fornecer parâmetros construtivos a um *applet* Java[20]. Apesar das métricas

[20] O *applet* Java utilizado foi criado pela empresa *Object Planet* (www.objectplanet.com).

calculadas serem posteriormente armazenadas no banco de dados, esta operação não é necessária para a construção dos gráficos.

A métrica IDC levanta informações das durações previstas no cronograma original e no último cronograma mestre inserido. O gráfico apresenta a razão entre estes dois subtotais, por status, conforme apresentado na Figura 59. O gráfico montado traz ainda uma linha de referência no valor igual a 1, valor este que referencia uma situação ideal segundo a qual não ocorrem desvios em relação ao cronograma planejado.

Figura 59 - Métrica IDC

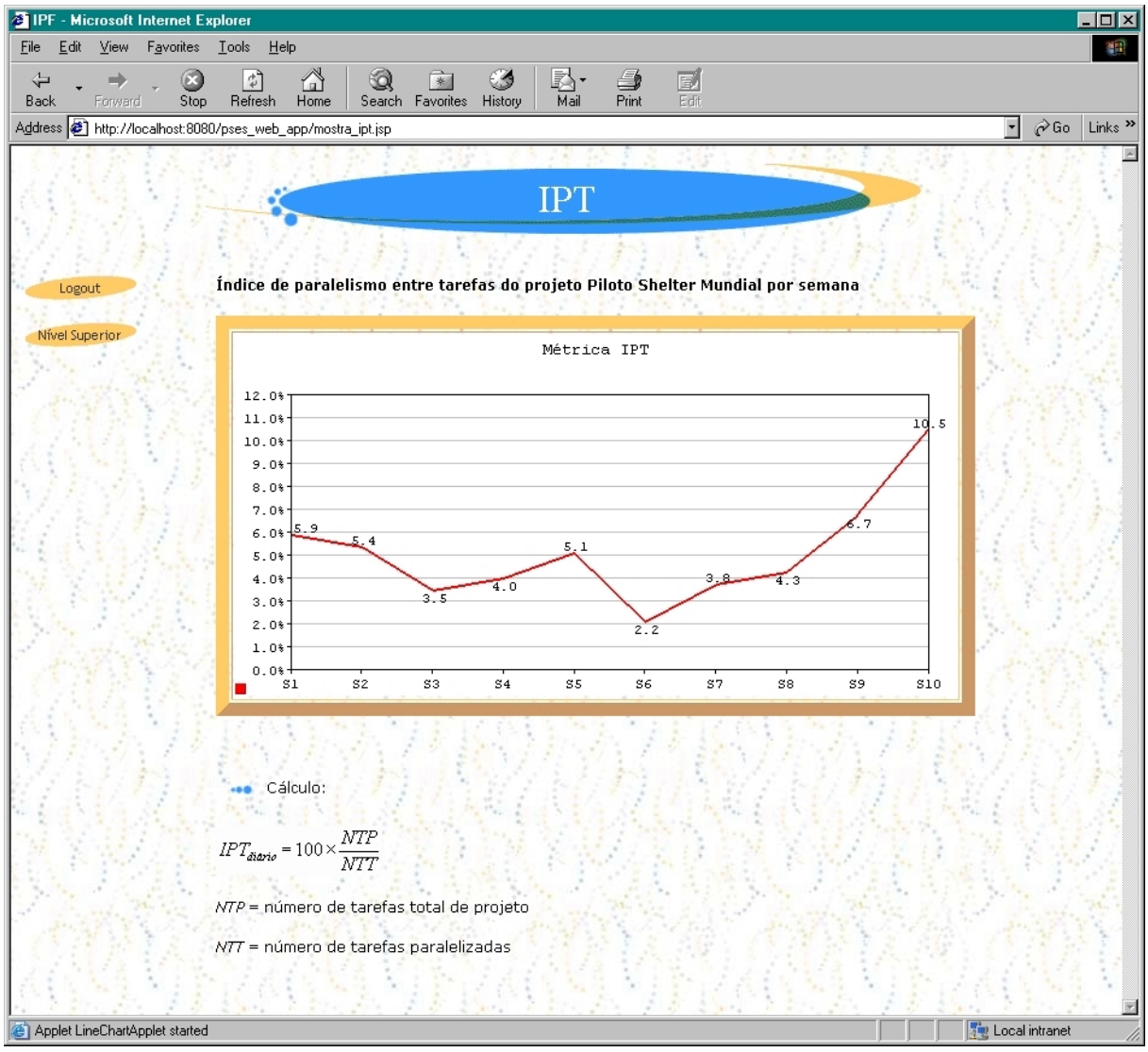


Percebe-se no exemplo ilustrado, que no momento em que o status B foi atingido, o cronograma mestre estava apresentando um desvio de 20% em relação ao

previsto, situação esta que foi amenizada quando o status C foi atingido, quando o desvio passou para 11% em relação ao previsto. Possivelmente tal melhora foi causada por intervenção da gerência de projeto no sentido de modificar a duração das tarefas através do ajuste de fatores como alocação de recursos humanos, dedicação, entre outros.

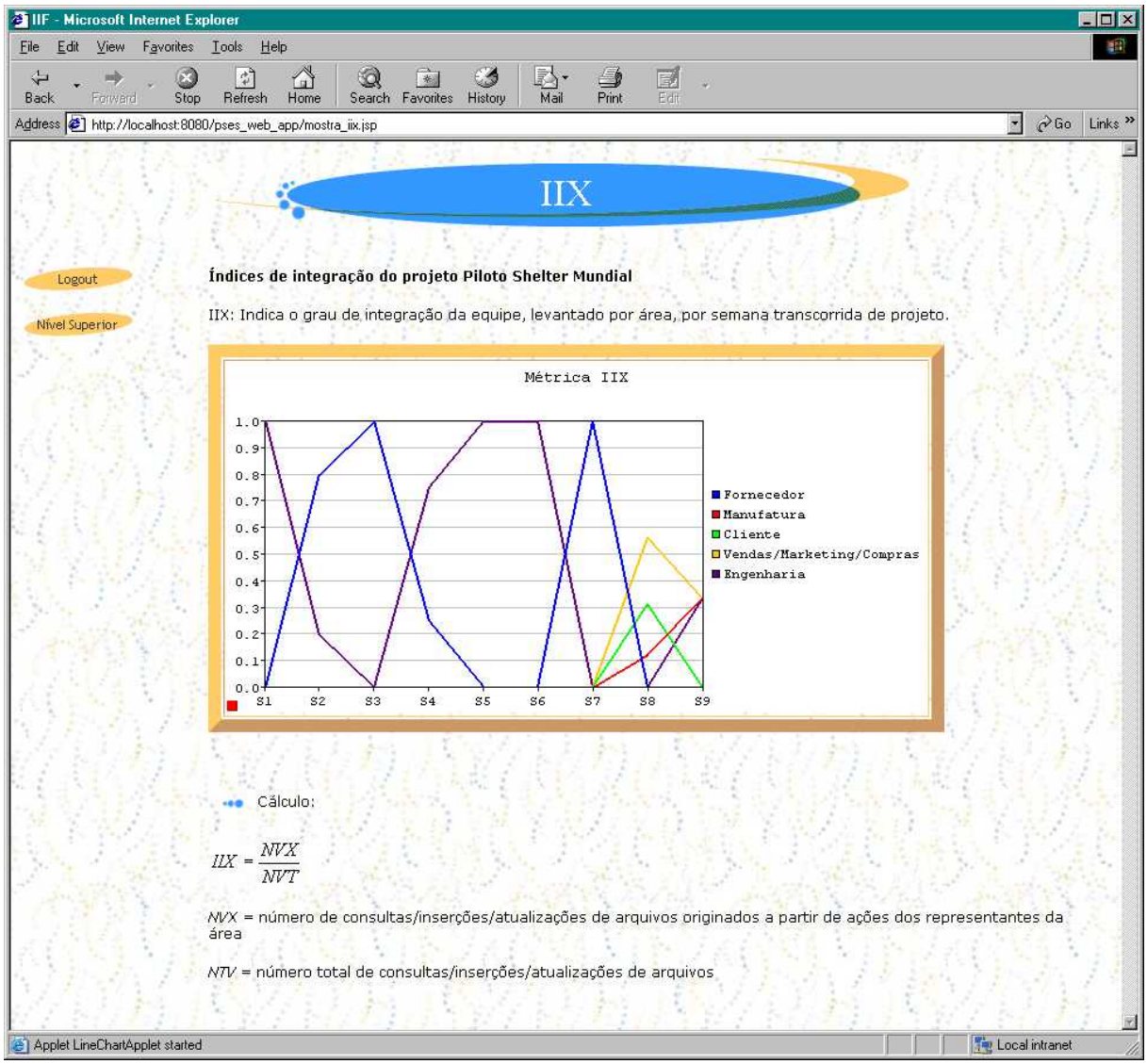
A métrica IPT calcula, por dia, qual é a razão entre o número de tarefas em paralelo e o número total de tarefas. Estes valores são então totalizados por semana transcorrida de projeto para melhor visualização. A Figura 60 traz um exemplo.

Figura 60 - Métrica IPT



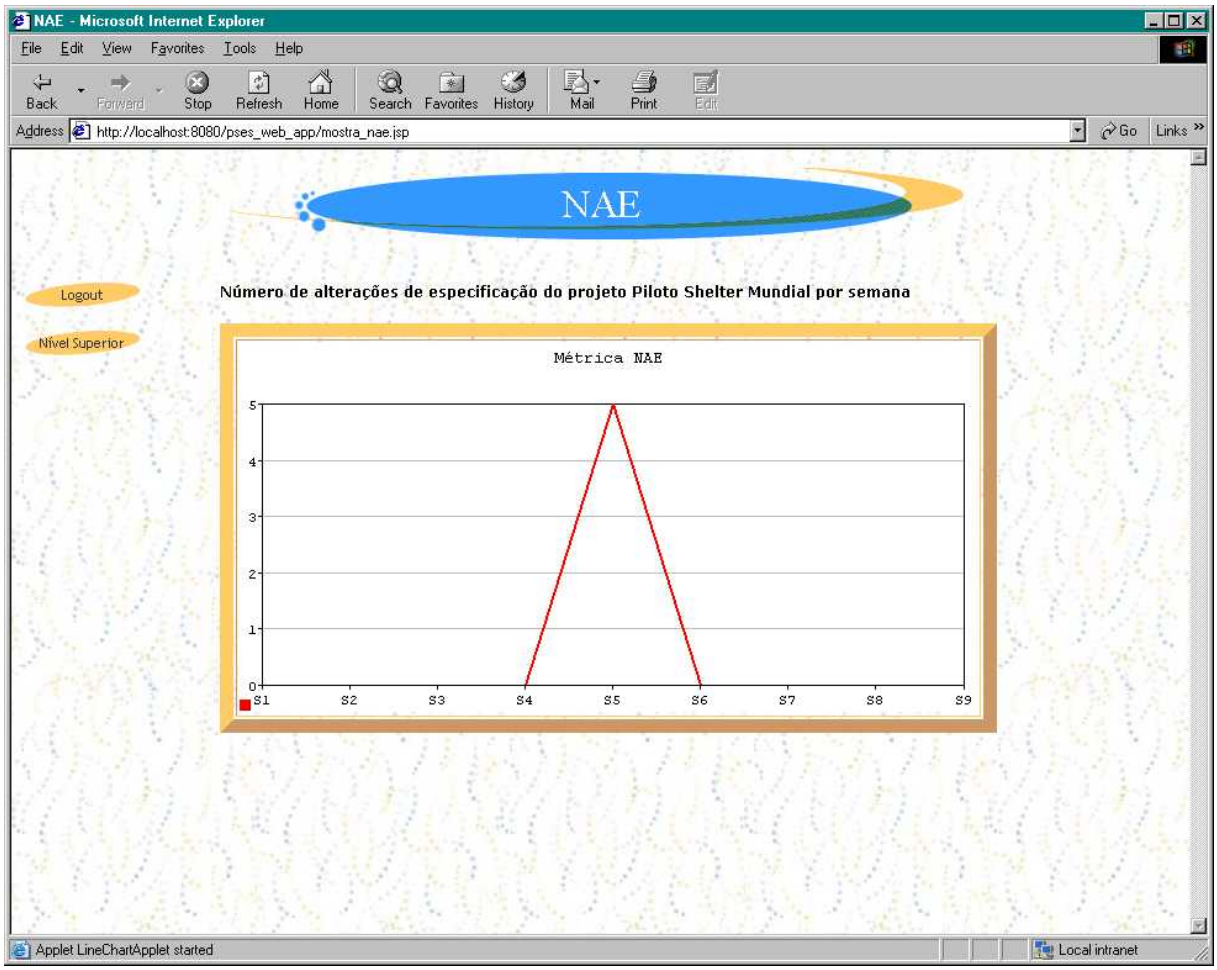
Nesta figura, percebe-se que, num cronograma padrão para este cenário, a sobreposição de atividades não passa de 11%. Mesmo assim, este valor pode ter sido atingido por interferência da gerência do projeto. É possível que para outros cenários os valores atinjam melhores marcas, lembrando que numa situação hipotética de sobreposição total das atividades, seria atingido o valor de 100% constante. Não foi encontrado na literatura nenhum valor de referência para esta métrica, o que sugere que estudos sejam conduzidos no sentido de fixar valores de referência a serem buscados em cada setor produtivo específico.

Figura 61 - Métrica IIX



A métrica IIX, por sua vez, apresenta a participação que cada área tem dentro da equipe de desenvolvimento, em função do número de intervenções de atualização, visualização e liberação de arquivos, por semana transcorrida de projeto. A Figura 61 apresenta um resultado.

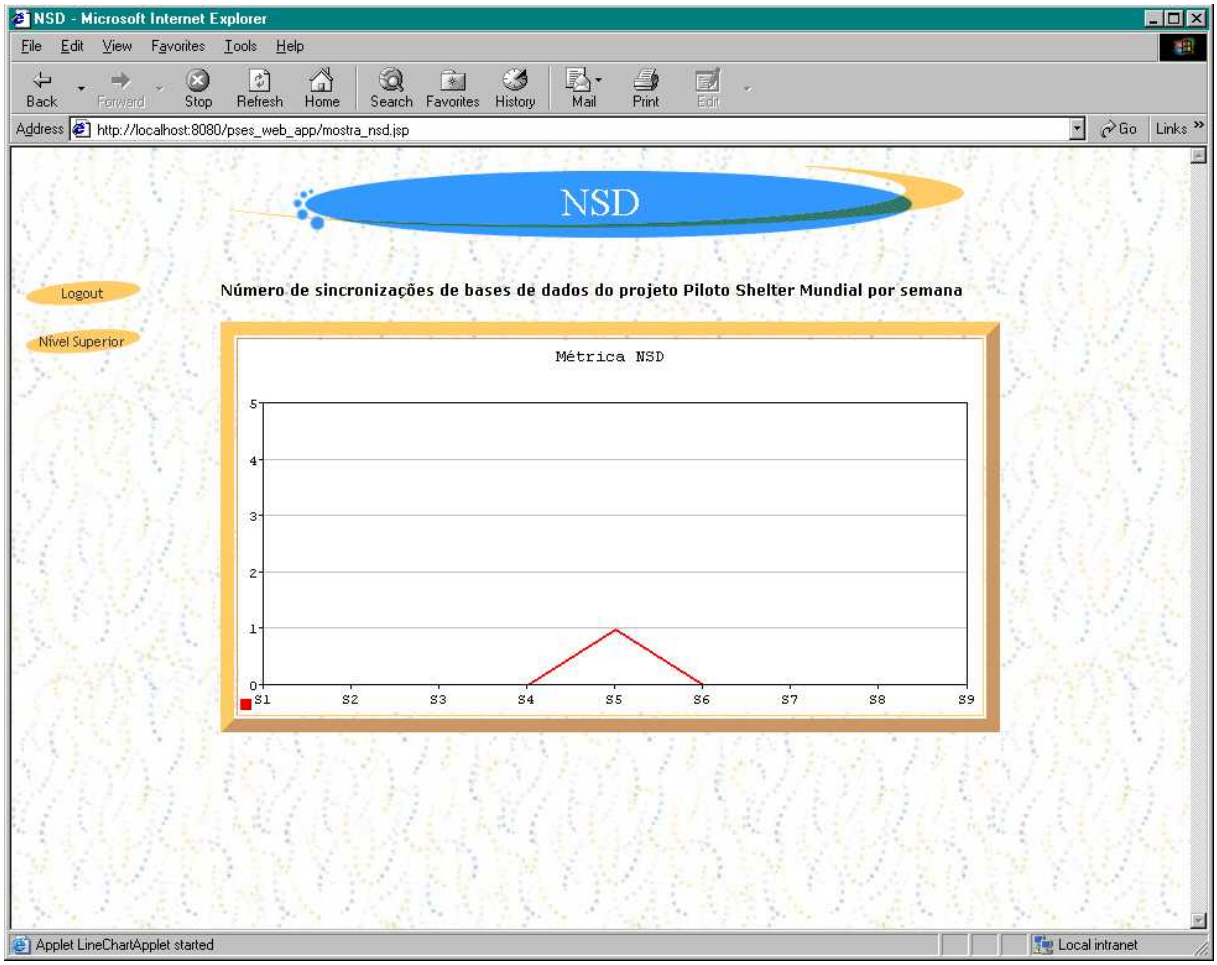
Figura 62 - Métrica NAE



Neste exemplo, cada linha traz a participação de áreas específicas tais como *marketing*, vendas e engenharia na construção do conhecimento que é agregado ao produto ao longo do tempo. Neste caso, percebe-se que há momentos em que apenas a área de engenharia contribui, porém o gráfico pode capturar a distribuição destas contribuições em outros momentos. Em tese, à medida que a equipe se aproxima da excelência na prática da ES, ocorreria uma distribuição mais equilibrada entre as

diferentes áreas de conhecimento dos colaboradores nas fases iniciais do projeto, representado a contribuição que cada área traria no sentido de minimizar correções posteriores.

Figura 63 - Métrica NSD



A métrica NAE apresenta, por semana transcorrida de projeto, a freqüência de atualizações dos arquivos que fazem parte do elemento mestre que contém as informações de especificação do produto. Para tanto, antes da apresentação da métrica, o sistema solicita ao gerente de projeto que identifique qual é o elemento onde constam tais informações. O sistema então apresenta uma lista dos arquivos do elemento e o gerente novamente identifica um ou mais arquivos que julga trazerem a essência da especificação do produto. A Figura 62 apresenta um exemplo do gráfico correspondente à métrica NAE.

Por último, a métrica NSD apresenta, de modo temporal, a forma como o modelo de informações sofreu sincronizações, através do levantamento de atualizações de *releases* efetuadas. A Figura 63 apresenta um destes resultados.

5 DISCUSSÃO

Durante o trabalho de modelagem desenvolvido, constatou-se que o levantamento dos procedimentos informalmente executados por projetistas e engenheiros de desenvolvimento, assim como demais profissionais envolvidos permitiu o surgimento de questionamentos entre gerentes e equipes e evidenciou falhas de comunicação. Foram percebidos alguns problemas que poderiam propiciar problemas de qualidade da informação que flui dentro das equipes e também oportunidades para que sessões de treinamento interno sejam conduzidas no sentido de orientar os participantes das equipes a executarem as tarefas conforme procedimentos já existentes, porém não completamente assimilados.

Como possível desdobramento da pesquisa realizada, poder-se-ia levantar dados como freqüência de ocorrência e tempo de execução de cada tarefa a fim de simular o ciclo de desenvolvimento e, em última instância, otimizá-lo. Decidiu-se que a otimização deste fluxo de atividades poderá ser realizada posteriormente em futuras investigações.

A descrição do modelo de informação também, por si só, seria suficiente para revelar possíveis problemas de troca de informação entre membros da equipe de desenvolvimento, na medida em que evidencia os vínculos entre seus componentes e impõe medidas no sentido de aproximar as pessoas que são afetadas por determinadas decisões de projeto, seja do ponto de vista de gestão ou detalhamento técnico. É possível determinar inconsistências na base de dados de projeto, o que poderia levar a problemas de qualidade em momentos posteriores do ciclo de desenvolvimento de novos produtos.

Através da realização desta pesquisa foi possível mensurar o grau de dificuldade para a concatenação das diversas abordagens da ES. Havia uma idéia inicial de se utilizar algumas ferramentas (PDM, CSCW, gestão de projetos, *workflow* e assim por diante) disponíveis atualmente no mercado para integrá-las e compor então um sistema maior, o que, se por um lado não exigiria um esforço de programação

muito grande, por outro lado necessitaria da exploração de todos os mecanismos de customização de tais ferramentas.

Esta estratégia se mostrou inviável, pois quando não haviam restrições orçamentárias para aquisição de ferramentas prontas, não era possível atingir o grau de customização desejado. E, para dificultar ainda mais a realização desta idéia preliminar, havia restrições impostas pelos fabricantes para o fornecimento, a título de pesquisa, de módulos de programação específicos para customização dos aplicativos. Não menos importante, mesmo que todas as dificuldades já relacionadas fossem transpostas, cada aplicativo necessitaria do domínio de uma linguagem de programação específica, tal como PHP, C, C++ e Java, entre outras.

Uma vez decidido pela implementação própria e definição das tecnologias a serem utilizadas, foi necessária uma fase de treinamento em temas como bancos de dados e linguagem SQL, *servlets* e JSPs, *applets* Java e, porque não mencionar, treinamentos nas ferramentas a serem utilizadas, como Together, JBuilder e PostgreSQL. São conhecimentos que formam a base para o desenvolvimento do sistema proposto e de futuras pesquisas.

A área de programação para ambiente de *internet* é extremamente dinâmica. Exemplo disso é que ao longo do desenvolvimento surgiram várias vezes questionamentos se uma ferramenta ou tecnologia deveria ser abandonada em favor de outra mais atualizada. O banco de dados **PostgreSQL**, por exemplo, teve o recurso de *pool* de conexões, essencial para escalabilidade do banco, incorporado apenas ao último *driver* JDBC disponibilizado recentemente. Optou-se nesta pesquisa pelas tecnologias consolidadas, de forma que aprimoramentos futuros sejam feitos com o uso de novas tecnologias, como XML, .NET, J2EE *Patterns*, entre outras, cuja curva de aprendizado seja mais acentuada.

O sistema PSES pode, no futuro, ser aprimorado em várias frentes. A primeira diz respeito à utilização de XML para substituir as complicadas topologias de bancos de dados relacionais. XML traz uma série de benefícios, porém um deles é muito interessante: a facilidade de compreensão dos relacionamentos entre

informações em função dos mesmos fazerem parte do esquema XML escolhido. Além disso, os bancos de dados tradicionais possibilitam hoje que a tecnologia XML seja utilizada diretamente.

Por outro lado, apenas recentemente o padrão XML passou a ser explorado em aplicativos comuns. Exemplo disso é a inclusão do padrão apenas na versão 2002 do aplicativo Microsoft Project. Mesmo assim, a sua utilização se mostrou bastante difícil, pois alterações externas no arquivo XML deste aplicativo não são reconhecidas se todos os demais componentes do arquivo não forem trabalhados igualmente. Desta forma o fabricante limita muito as ações de desenvolvedores independentes, que procuram criar sistemas integrados ao aplicativo.

Há, entretanto, iniciativas promissoras no sentido de se criar um esquema XML específico para a área de gestão de projetos como o PMXML, onde PM significa *Project Management*. Através de um consórcio de empresas interessadas num padrão universalizante, já existe uma proposta de esquema disponível, que infelizmente ainda não está sendo utilizada pela Microsoft. Mesmo que o PMXML seja aproveitado em desenvolvimentos futuros do sistema PSES, ele deverá ser estendido para contemplar informações de projeto e não apenas tarefas e recursos. Em síntese, o trabalho para desenvolvimento do DER no sistema atual deve ser aproveitado diretamente na concepção de um novo esquema XML, que deve ainda levar em conta o padrão PMXML (PACIFIC EDGE SOFTWARE, 2003).

O modelo de navegação do sistema desenvolvido neste trabalho é atualmente denominado de **modelo 1**. Neste modelo ocorre um cascateamento de páginas JSP que implica num grande número de páginas e dificuldade de navegação. Uma possibilidade seria reescrever o atual sistema utilizando o **modelo 2**, segundo o qual *servlets* são utilizados como centralizadores para o redirecionamento de páginas JSP (HALL, 2000). Esta modificação produziria um sistema de navegação mais amigável.

O sistema atual foi desenvolvido com um certo grau de independência entre *design* das páginas e lógica de programação. A larga utilização de *scriplets* Java dentro do código HTML torna difícil a manutenção das páginas. A utilização de *tags*

customizados poderia reduzir este problema, permitindo que equipes totalmente distintas executassem o *design* e a implementação da lógica propriamente dita (HALL, 2002).

Outra possibilidade concreta de melhoria no sistema atual é a utilização da especificação J2EE e o uso de EJBs. Isto significaria que o sistema passaria a ser um **sistema distribuído** ou **de objetos distribuídos**, onde o uso de componentes (EJBs) reaproveitáveis traria ainda benefícios como segurança, desempenho, compatibilidade e facilidade de manutenção (KURNIAWAN, 2002; ROMAN; AMBLER; JEWELL, 2002). EJBs delegam a entidades chamadas **contêiners EJBs** operações como persistência e comunicação cliente/servidor. Por outro lado, novos servidores são necessários, o que adiciona complexidade à arquitetura do sistema.

Com relação às funcionalidades propostas nas especificações (ver 3.4.4), praticamente todas puderam ser implementadas no sistema PSES. Entretanto, a proteção aos dados que trafegam na rede através de criptografia e os mecanismos de *backup* não foram implementados. O recurso de criptografia poderia ser implementado se houvesse disponibilidade de classes Java específicas associadas à transferência de arquivos via protocolo FTP. As classes Java utilizadas, fornecidas pela empresa IPWorks, a título de experimentação junto com a ferramenta JBuilder, não trazem estes recursos.

Quanto ao *backup*, o administrador de sistema pode lançar mão de recursos do próprio sistema operacional para armazenar o conteúdo dos repositórios distribuídos. Seria interessante, entretanto, que houvesse algum dispositivo que permitisse a automatização deste procedimento, a ser disparado a partir do servidor *Web* do sistema.

Um outro ponto de aperfeiçoamento do sistema atual é a incorporação de um módulo para inserção de cenários. Os cenários, para efeito de prova de conceito, foram inseridos de forma manual. Tal procedimento consistiu na elaboração de um cronograma mestre em XML, um modelo de informações descrito por *scripts* SQL e vários JSPs contendo o *workflow*. Após a inserção, o sistema deve ser recompilado.

Em se tratando da atual arquitetura do sistema, o processo de recompilação das páginas de um novo cenário pode ser feita sem a desativação do contêiner JSP, entretanto não foram feitos testes de recompilação do sistema com vários usuários logados simultaneamente.

Num cenário real de desenvolvimento de produtos utilizando preceitos da ES, é geralmente há muitas iterações ao longo do projeto. Para uma situação de realização de tarefas em **paralelo**, mas de forma **linear**, o sistema se comporta de forma satisfatória. Caso sejam implementados os recursos de desvios (*go-to*) presentes no fluxo (atualmente utilizados apenas como referência), e possível necessidade de redefinição dos percentuais completados das tarefas dependentes, haveria questões que teriam de ser tratadas, tais como a geração de eventos, reinicialização de tarefas anteriormente dadas por completas e assim por diante. Os problemas causados por iterações profundas, ou seja, que provocam a reconsideração de muitas tarefas, ainda teria de ser analisado com cuidado. A princípio tais problemas devem ser solucionados sem a modificação de arquitetura ou tecnologias adotadas atualmente, bastando apenas algum esforço de programação.

Outra possibilidade levantada ao longo do desenvolvimento do sistema PSES é a de se abandonar o conceito de *workflow*, pelo menos da forma como é visto hoje, em favor de uma visão de processo mais inovadora. Seria, em tese, a utilização das tarefas não como atividades passivas que devam ser realizadas conforme um "mapa", mas sim como agentes dinâmicos que negociem entre si de forma pro-ativa a sua própria realização e, através de eventos internos, disparem ações de notificação, bloqueio, comunicação, e assim por diante. Tal concepção seria implementada através de sistemas baseados em **agentes**, em contraste aos sistemas baseados em **objetos** (BARTHÈS e SHEN, 2002; SCHIMMECK, 2002). Uma concepção como esta geraria controvérsia, ao apontar para uma direção contrária à adoção de modelos de referência, cuja repetibilidade seja uma característica valorizada. Atualmente a previsibilidade de processos é buscada através da aplicação dos preceitos da ES. Hipoteticamente, processos montados dinamicamente poderiam servir para casos bastante específicos de

desenvolvimento de produtos.

Apesar de bastante simples, o recurso de painel de discussão (*whiteboards*) adotado do sistema PSES poderia ser ampliado para contemplar novas possibilidades de colaboração via voz, vídeo e *application sharing*. Entretanto, o conceito de associação de discussões ao *workflow* e registro junto ao modelo de informações, tal como ocorre no sistema atual deve ser mantido. O sistema carece de um mecanismo de busca que permita resgatar discussões contextualizadas que tenham ocorrido ao longo do processo de desenvolvimento, para que o conhecimento gerado não se perca. Atualmente muito do que se constrói em nível de conhecimento no DIP não fica registrado. Entretanto, não basta apenas resgatar os dados no sistema. Algum tratamento deve ser concebido, talvez através de sistemas especialistas.

A contribuição da presente pesquisa pode ser compreendida em três dimensões distintas. Primeiramente, foi desenvolvida uma sistemática que permite capturar cenários de DIP levando-se em consideração tanto o processo quanto o teor das informações que são agregadas nas diferentes tarefas que ocorrem, bem como a maneira pela qual os elementos de informação se relacionam uns aos outros. Ao mesmo tempo em que os cenários podem contemplar as particularidades de um produto e de uma empresa, permitem que o processo possa ser repetido e estimado, sem surpresas além dos riscos tecnológicos e mercadológicos naturais do mesmo.

Em segundo lugar, a sistemática proposta permite que sistemas de informação sejam mais precisos e demonstrem aderência ao problema do qual pretendem tratar. O DIP é uma questão de integração entre pessoas e qualquer sistema que se proponha a facilitá-lo deve prever as transações que ocorrem.

Finalmente, houve esta pesquisa o propósito de se utilizar sínteses de todas as abordagens relevantes presentes na literatura a respeito de ES. Nem por isso o assunto está esgotado. Faltava uma tentativa no sentido de reunir numa única plataforma todas as perspectivas que existem sobre o assunto.

É importante registrar que a tese de que um sistema de informação poderia ser desenvolvido para contemplar as diferentes abordagens de um cenário de DIP é

verdadeira. Entretanto, é necessário que testes de teor prático sejam conduzidos de forma que não só modelos de processos e informação e sistema de métricas sejam validados, mas também desempenho, facilidade de uso e versatilidade do sistema proposto sejam avaliados. Com base nestas avaliações pode-se optar por novas tecnologias ou mesmo arquiteturas para o sistema. Tal como é feito em pesquisas para a compreensão da problemática social e técnica do DIP (SMITH e TJANDRA, 1998), uma avaliação do sistema PSES através de jogos de papéis do tipo RPG pode ser apropriada em futuras pesquisas.

6 CONCLUSÃO

Neste trabalho de investigação, objetivou-se desenvolver uma estrutura de organização de trabalho para o DIP, baseada em preceitos da ES. Como parte integrante da estrutura de trabalho almejada, a pesquisa tinha como meta a definição de modelos de informação e processo, assim como um sistema de métricas e, finalmente, um sistema de informações que, baseado em tais modelos, atuasse como elemento facilitador para a aplicação de conceitos da ES.

Como resultado final da pesquisa, foi possível definir uma forma para capturar modelos referenciais de processos e informações em cenários de DIP, assim como propor um conjunto de métricas passível de utilização direta num sistema de informações. Tal sistema, denominado PSES, foi construído com componentes conhecidos de tecnologia da informação e cumpre uma série de requisitos definidos *a prori*, a fim de servir à situação de desenvolvimento distribuído de produtos de alto teor tecnológico. O sistema PSES, apresenta características que permitem a integração inédita de funcionalidades desejáveis derivadas de sistemas do tipo PDM, *Workflow*, CSCW e gestão de projeto, reunidas num único sistema que acessa uma base de dados compartilhada. O conjunto de técnicas para modelagem de cenários, modelos propriamente ditos, sistema de métricas e sistema de informação propostos nesta investigação fornece uma estrutura de trabalho objetivada para o DIP.

No modelo de processo, ocorrem as definições de quais atividades devem ser conduzidas e quando, qual seu nível de interdependência e quais informações são trabalhadas. No modelo de informação, são definidas as vinculações entre os diferentes elementos de informação e de que forma tais elementos contemplam a decomposição do produto em subsistemas e componentes. Também neste modelo é descrita a forma como as informações devam ser agregadas ao produto até que se chegue ao ponto de liberação para produção. Foram utilizadas técnicas de captura para este modelo (IDEF3) e o aproveitamento dos mesmos num sistema de informações. Um sistema de métricas foi definido de forma que os aspectos mais relevantes para a

avaliação da prática dos preceitos da ES foram contemplados. O sistema de métricas também foi preparado para sua implementação direta no sistema de informações proposto.

Após o levantamento de requisitos do sistema de informações com referência às diferentes abordagens da ES, a implementação do sistema PSES foi iniciada através de modelagem UML dos casos de uso na ferramenta **Together**. A base de dados foi modelada com a ferramenta **ERwin** através de um DER, diagrama este aproveitado para a geração de *scripts* SQL para criação de uma base de dados em **PostgreSQL**. Com base nestes modelos, a codificação do sistema foi feita com a ferramenta **JBuilder**, utilizando tecnologia Java (*applets*, JDBC, JSP e *Javabeans*).

O aproveitamento do modelo de processo se deu através da conversão dos modelos IDEF3 capturados através da ferramenta **ProSim** em cronogramas de projeto em formato XML utilizando a ferramenta **Microsoft Project 2002** e *workflows* sob a forma de *applets* Java em JSPs. O modelo de informação foi convertido em *scripts* SQL que alimentam tabelas de entidades previstas no DER. O sistema de métricas foi implementado com o uso de *applets* desenvolvidos por terceiros que formatam dinamicamente dados resgatados do banco de dados sob a forma de gráficos de linhas. Os recursos de *upload/download* de arquivos do segmento PDM do sistema foram implementados com o uso de classes de desenvolvedores independentes para a transferência de arquivos via protocolo FTP. Recursos de correio eletrônico e relação de eventos de projeto foram explorados para notificação dos colaboradores (*awareness*). O sistema desenvolvido é um protótipo conceitual, que se encontra em situação operante.

Apesar de se ter chegado num ponto no qual é possível afirmar que os objetivos da pesquisa foram alcançados, há uma série de questionamentos que surgem em função do mesmo não ter sido ainda aplicado a situações reais de DIP. O mais importante questionamento, portanto, é se tal sistema será efetivamente um elemento facilitador para equipes de DIP. Conclusões somente poderão ser tiradas a partir do momento em que os indicadores do sistema, além de outros mecanismos de avaliação

de desempenho, tais como novos questionários aplicados aos usuários, sejam aplicados e interpretados.

Da mesma forma, no atual estágio de desenvolvimento, ainda não se sabe qual será o nível de aceitação de um sistema tal como o proposto neste trabalho por equipes de DIP. Sistemas de informação serão tão bem aceitos quanto mais se aproximarem da realidade social dos indivíduos que o utilizam (EASON, 1988). Portanto, apesar de terem sido tomados cuidados para que os modelos fossem criados a partir de uma imagem fiel do processo social que acontece numa equipe de DIP para um determinado cenário, é necessário avaliar a percepção que os usuários terão quando utilizarem o sistema em seu dia-a-dia. Nesta mesma linha, outros questionamentos podem ser feitos: seria o nível atual de detalhamento dos modelos de processo e informação adequados aos diferentes níveis de abstração assumidos pelos membros da equipe de DIP? Ou: o levantamento de métricas efetuado diretamente pelo sistema possibilitará tomadas de decisão mais precisas e úteis? E, ainda: o desempenho do sistema, relacionado com a arquitetura adotada, atenderá as necessidades de equipes de DIP geograficamente distribuídas? A resposta só poderá surgir após uma avaliação da utilização do sistema proposto.

Com relação à evolução do sistema PSES, foram propostos aperfeiçoamentos e testes nesta pesquisa. Dentre as possibilidade de aperfeiçoamento indicados destacam-se a inclusão de mecanismos para facilitar a inserção de cenários, o desenvolvimento de uma lógica de programação que possibilite a realização de interações controladas pelos recursos de *workflow* do sistema, a modificação da arquitetura do sistema através da adoção da especificação J2EE e a criação de uma versão do sistema baseada em agentes. Sugeriu-se que testes através de RPG sobre situações hipotéticas de desenvolvimento sejam aplicados para avaliações preliminares do desempenho e versatilidade do sistema.

REFERÊNCIAS

1 AITSAHLIA, F.; JOHNSON, E.; WILL, P. Is Concurrent Engineering Always a Sensible Proposition? **IEEE Transactions on Engineering Management**, v. 42, n. 2, p. 166-170, May 1995.

2 AMBIÊNCIA SISTEMAS DE INFORMAÇÃO LTDA. **ProSLCSE/Fun®**. Disponível em: <http://www.ambiencia.com.br> Acesso em: 17 abr. 2003. Ferramenta de modelagem e simulação de processos de negócios.

3 ANDERSON, D. M. **Agile Product Development For Mass Customization, JIT, Build-to-Order, and Agile Manufacturing**. Disponível em: < http://www.soce.org/papers/dma-agile/agile_prod_dev.htm> Acesso em: 13 mar. 2003.

4 ANSALDI, S.; GIANNINI, F. On Data Model Communication in Design Activities. In: INTERNATIONAL SYMPOSIUM ON TOOLS AND METHODS FOR COMPETITIVE ENGINEERING 2000: Delft. **Proceedings of the Third International Symposium on Tools and Methods for Competitive Engineering**, Delft, 2000. p.203-216.

5 ANUPAM, V.; BAJAJ, C. L. Shastra: Multimedia Collaborative Design Environment. **IEEE MultiMedia**, p. 39-49, Summer 1994.

6 APACHE SOFTWARE FOUNDATION. **Tomcat®**. Disponível em <http://jakarta.apache.org/tomcat/index.html> Acesso em: 22 abr. 2003. 1999. Contêiner servlet.

7 BAAKE, U.; HAUßMANN, D.; STRATIL, P. Management of Concurrent Engineering Processes in international product development. In: 5TH ISPE INTERNATIONAL CONFERENCE ON CONCURRENT ENGINEERING RESEARCH AND APPLICATIONS 1998: Tóquio. **Advances in Concurrent Engineering**. Tóquio, 1998. p. 145-154.

8 BAILEY, J. E.; RUCKER, R. H. A Web-Based System to Enhance IPPD by Automating Designer Communication and Data Access. In: USHER, JOHN M. ROY, UTPAL. PARSAEI, HAMID R. (Ed). **Integrated Product and Process Development - Methods, Tools, and Technologies**. New York, NY: John Wiley & Sons, 1998. p.125-148.

9 BARTHÈS, J. P.; SHEN. W. **Introduction to Agent Technology and Examples of Applications in Engineering Design**. Compiègne: UTC/NRC-CNRC, 2002. 104 transparências: p&b.

10 BLACKBURN, J. D.; SCUDDER, G. D.; WASSENHOVE, L. N. V.; HILL, C. Time-based software development. **Integrated Manufacturing Systems**, v. 7, n. 2, p. 60-66, 1996.

11 BLACKBURN, J. D; HOEDEMAKER, G.; WASSENHOVE, L. N. V. Concurrent Software Engineering: Prospects and Pitfalls. **IEEE Transactions on Engineering Management**, v. 43, n. 2, p. 179-188, May 1996.

12 BORLAND SOFTWARE CORPORATION. **JBuilder®. Version 4.0**. 2000. 1 CD-ROM. Ferramenta IDE de programação em Java.

13 BORLAND SOFTWARE CORPORATION. **JBuilder®. Version 6.0**. 2002a. 1 CD-ROM. Ferramenta IDE de programação em Java.

14 BORLAND SOFTWARE CORPORATION. **Together®. Version 6.0**. 2002b. 1 CD-ROM. Ferramenta de modelagem e programação OO.

15 CASAROTTO FILHO, N.; FÁVERO, J. S.; CASTRO, J. E. E. **Gerência de Projetos/Engenharia Simultânea**. São Paulo: Editora Atlas, 1999.

16 CHEE, C. F. Y.; GOH, A. Implementing Concurrent Engineering Design Process. In: 5TH ISPE INTERNATIONAL CONFERENCE ON CONCURRENT ENGINEERING RESEARCH AND APPLICATIONS 1998: Tóquio. **Advances in Concurrent Engineering**. Tóquio, 1998. p. 425-431.

17 CLARK, K. B.; FUJIMOTO, T. The Power of Product Integrity. In: CLARK, K. B.; WHEELWRIGHT, S. C. (Ed.). **The Product Development Challenge**. Boston: Harvard Business Review, 1995. p.277-296.

18 CLAUSING, D. Concurrent Engineering. **ASM Handbook**, v. 20, 1997.

19 CLEETUS, K. J. **Definition of Concurrent Engineering - CERC Technical Report Series, Research Note, CERC-TR-R-92-003**. Morgantown, WV: Concurrent Engineering Research Centre, West Virginia University, 1992. p. 1-5. Relatório técnico.

20 COOPER, R. G.; KLEINSCHMIDT, E. J. Uncovering the Keys to New Procduct Success. **Engineering Management Review**, p. 5-18, Winter 1993.

21 CRABTREE, R. A.; FOX, M. S.; BAID, N. K. Case Studies of Coordination Activities and Problems in Collaborative Design. **Research in Engineering Design**, n. 9, p. 70-84, 1997.

22 CRAWFORD, C. M. The Hidden Costs of Accelerated Product Development. **Journal of Product Innovation Management**, v. 9, n. 3, p. 188-189, Sept. 1992.

23 CROW, K. **Customer-focused Development with QFD**. Disponível em: <http://www.soce.org/papers/dma-agile/agile_prod_dev.htm> Acesso em: 19 mar. 2003.

24 CRUZ, T. **Workflow: A Tecnologia que Vai Revolucionar Processos**. São Paulo: Editora Atlas, 1998.

25 DANGELMAIER, W.; KRESS, S.; WENKI, R. TelCoW: telework under the co-ordination of a workflow management system. **Information and Software Technology**, v. 41, p. 341-353, 1999.

26 DARNTON, G. Working together: a management summary of CSCW. **Computing & Control Engineering Journal**, p. 37-42, Feb. 1995.

27 DBEXPERTS. **PostgreSQL®. Versão 7.1.2**. São Paulo, 2001. 1 CD-ROM. Banco de dados relacional.

28 DOCHERTY, M. Pre-Conference Workshop. In: PRODUCT & PROCESS LEADERSHIP WEEK 1998: Boston, MA. **Project Management for Rapid Product Development**. Boston, MA: The Management Roundtable/Creative Logic, Inc., 1998.

29 DOMAZET, D. S.; CHOONG, F. N.; SNG, D.; HO, N. C.; LU, S. C. Y. Active Data-Driven Design Using Dynamic Product Models. **Annals of the CIRP**, v. 44, n. 1, p. 109-112, 1995.

30 DOWLATSHAHI, S. A Comparison of Approaches to Concurrent Engineering. **International Journal of Advanced Manufacturing Technology**, v. 9, p. 106-113, 1994.

31 DWIVEDI, S. N.; KULPA, Z.; SOBOLEWSKI, M. Graphical and natural language interface with a knowledge-based concurrent engineering environment. **Computers in Industry**, v. 23, n. 3, p. 175-184, Dec. 1993.

32 EASON, K. **Information Technology and Organisational Change**. London: Taylor & Francis, 1988.

33 EHRLENSPIEL, K.; GIAPOULIS, A.; GÜNTHER, J. Teamwork and Design Methodology - Observations About Teamwork in Design Education. **Research in Engineering Design**, v. 9, p. 61-69, 1997.

34 ELLIOT, S. S.; LASSER, R. Pre-Conference Workshop. In: DESIGN @NYWHERE MANUFACTURE @NYWHERE 2: Tampa, FL. **Leading Co-Wired Teams**. Waltham, MA: The Management Roundtable, 2001.

35 EPPINGER, S. D.; NUKALA, M. V.; WHITNEY, D. E. Generalised Models of Design Iteration Using Signal Flow Graphs. **Research in Engineering Design**, v. 9, p. 112-123, 1997.

36 EPPINGER, S. D.; WHITNEY, D. E.; SMITH, R. P.; GEBALA, D. A. A Model-Based Method for Organizing Tasks in Product Development. **Research in Engineering Design**, v. 6, p. 1-13, 1994.

37 EPPINGER, S. Minicurso. In: DESENVOLVIMENTO RÁPIDO DE PRODUTOS TECNOLÓGICOS: Florianópolis. **Accelerating Complex Product Development Process**. Florianópolis: Fundação Certi/MIT Sloan, 1998.

38 EVBUOMWAN, N. F. O.; ANUMBA, C. J. Towards a concurrent engineering model for design-and-build projects. **The Structural Engineer**, v. 74, n. 5/5, p. 73-78, Mar. 1996.

39 FRANKENBERGER, E.; AUER, P.. Standardized Observation of Team-work in Design. **Research in Engineering Design**, v. 9, p. 1-9, 1997.

40 GARDINER, G. Concurrent and systems engineering: same thing, different name, or are they both just new product introduction? **Engineering Management Journal**, p. 17-22. Feb. 1996.

41 GATENBY, D. A. Concurrent Engineering: An Enabler For Fast, High-Quality Product Realization. **AT&T Technical Journal**, p. 34-47, Jan./Feb. 1994.

42 GOLDENSE, B. L. **Concurrent Product Development**. Cambridge, MA: Goldense Group Inc., 1997a. 290 transparências: p&b.

43 GOLDENSE, B. L. **Measuring Product Development**. Cambridge, MA: Goldense Group Inc., 1997b. 169 transparências: p&b.

44 HA, A. Y.; PORTEUS, E. L. Optimal Timing of Reviews in Concurrent Design for Manufacturability. **Management Science**, v. 41, n. 9, p. 1431-1447, Sept. 1995.

45 HALL, M**. Servlets & JavaServer Pages™ Training Course**. Upper Saddle River, NJ: Prentice Hall, 2000.

46 HANDFIELD, R. B. Effects of Concurrent Engineering on Make-to-Order Products. **IEEE Transactions on Engineering Management**, v. 41, n. 4, p. 384-393, Nov. 1994.

47 HARTLEY, J. R. **Engenharia Simultânea**. São Paulo: Editora Bookman, 1998.

48 HAUPTMAN, O. The Influence of Process Concurrency on Project Outcomes in Product Development: An Empirical Study of Cross-Functional Teams. **IEEE Transactions on Engineering Management**, v. 43, n. 2, p. 153-163, mai. 1996.

49 HULL, F. M.; COLLINS, P. D.; LIKER, J. K. Composite Forms of Organization as a Strategy for Concurrent Engineering Effectiveness**. IEEE Transactions on Engineering Management**, v. 43, n. 2, p. 133-142, May 1996.

50 HUNDAL, M. S. Time-Driven Product Development. In: USHER, JOHN M. ROY, UTPAL. PARSAEI, HAMID R. (Ed). **Integrated Product and Process Development - Methods, Tools, and Technologies**. New York, NY: John Wiley & Sons, 1998. p.59-84.

51 HUTHWAITE, B. **Strategic Design: A Guide to Managing Concurrent Engineering**. Chicago, MI: Institute for Competitive Design, 1994.

52 INTERNATIONAL SOCIETY FOR PRODUCTIVITY ENHANCEMENT (ISPE). **The International Institute of Concurrent Engineering**. Disponível em <http://www.secs.oakland.edu/SECS_prof_orgs/ISPE/index.html> Acesso em: 23 abr. 2003.

53 KANNAPAN, S. M.; MARSHEK, K. M. A scheme for negotiation between intelligent design agents in concurrent engineering. **Intelligent Computer Aided Design**, v. 4, Cap. 36, p. 1-5, 1992.

54 KAPPEL, T. A.; RUBENSTEIN, A. H. Creativity in Design: The Contribution of Information Technology. **IEEE Transactions on Engineering Management**, v. 46, n. 2, p. 132-143, mai. 1999.

55 KAYDOS, W. **Operational Performance Measurement - Increasing Total Productivity**. Boca Raton, FL: St. Lucie Press, 1999.

56 KERZNER, H. **Project Management: a systems approach to planning, scheduling and controlling**. New York, NY: International Thomson Publishing, 1995.

57 KHURANA, A.; ROSENTHAL, S. R. Integrating the Fuzzy Front End of New Product Development. **IEEE Engineering Management Review**, p. 35-49, Winter 1997.

58 KIM, H.; CHUNG, M. J. A Web-based Framework for Engineering Design Process in a Concurrent Engineering Environment. In: 5TH ISPE INTERNATIONAL CONFERENCE ON CONCURRENT ENGINEERING RESEARCH AND APPLICATIONS 1998: Tóquio. **Advances in Concurrent Engineering**. Tóquio, 1998. p. 417-423.

59 KING, N.; MAJCHRZAK, A. Concurrent Engineering Tools: Are the Human Issues Being Ignored? **IEEE Transactions on Engineering Management**, v. 43, n. 2, p. 189-201, May 1996.

60 KNOWLEDGE BASE SYSTEMS, INC. **ProSim® Automated Process, Cost Analysis, and Simulation Modeling for Windows®. Version 6.0**. Austin, TX, 2000a. 1 CD-ROM. Ferramenta de modelagem e simulação de processos de negócios.

61 KNOWLEDGE BASE SYSTEMS, INC. **ProSim® Option Pack - Automated Extensions to ProSim®. Version 2.0**. Austin, TX, 2000b. 1 CD-ROM. Pacote de módulos para complementação da funcionalidade do aplicativo ProSim.

62 KURNIAWAN, Budi. **Java for the Web with Servlets, JSP, and EJB - A Developer's Guide to Scalable J2EE Solutions**. Indianapolis, IN: New Riders, 2002.

63 KYRATSIS, E. P.; MANSON-PATRIDGE, B. M. Implementing Concurrent Engineering in Supply Chain Companies and the role of CAE Technology. **Engineering Designer**, p. 4-8, Sept./Oct. 1999.

64 LIANG, T.; LAI, H.; CHEN, N. C. When Client/Server Isn´t Enough: Coordinating Multiple Distributed Tasks. **IEEE Computer**, p. 73-79, May 1994.

65 LOCH, C.; STEIN, L.; TERWIESCH, C. Measuring Development Performance in the Electronics Industry. **The Journal of Product Innovation Management**, v. 13, n. 1, p. 3-20, Jan. 1996.

66 LOGIC WORKS, INC. **ERwin®. Version 3.5.2**. Princeton, NJ, 1997. 1 CD-ROM. Ferramenta de modelagem de dados.

67 LOVE, N. **Better Designs, Faster Development**. Disponível em: <http://www.soce.org/papers/lbl_consult/better_faster_dev.htm> Acesso em: 13 mar. 2003.

68 MAHER, M. L.; RUTHERFORD, J. H. A Model for Synchronous Collaborative Design Using CAD and Database Management. **Research in Engineering Design**, v. 9, p. 85-98, 1997.

69 MANSFIELD, T.; KAPLAN, S.; FITZPATRICK, G.; PHELPS, T.; FITZPATRICK, M.; TAYLOR, R. Toward Locales Supporting Collaboration with Orbit. **Information and Software Technology**, v. 41, p. 367-382, 1999.

70 MAYER, R.J.; MENZEL, C. P.; PAINTER, M. K.; DEWITTE, P. S.; BLINN, T.; PERAKATH, B. **Information Integration for Concurrent Engineering (IICE) IDEF3 Process Description Capture Method Report - Interim Technical Report for Period April 1992-September 1995**. Disponível em: <http://www.idef.com/idef3.html> Acesso em: 17 abr. 2003, 1995. Relatório técnico.

71 MELLO, S. A Conference Introduction. In: PRODUCT & PROCESS LEADERSHIP WEEK 1997: Boston, MA. **Product Development Basics**. Boston, MA: The Management Roundtable/Product Development Consulting, Inc., 1997.

72 MEYER. The development manufacturing interface: Empirical analysis of the 1990 European manufacturing futures survey. In: SUSMAN, G. I. (Ed**.**)**. Integrating Design and Manufacturing for Competitive Advantage**, New York, NY: Oxford, 1990.

73 MICROSOFT CORPORATION. **Microsoft FrontPage® 1998 for Windows®**. Redmond, WA, 1998. 1 CD-ROM. Ferramenta de diagramação de páginas HTML.

74 MICROSOFT CORPORATION. **Microsoft Project® 2002 for Windows®. Beta Release**. Redmond, WA, 2002. 3 CD-ROMs. Ferramenta de gestão de projetos.

75 PAASHUIS, V.; BOER, H. Organizing for concurrent engineering: an integration mechanism framework. **Integrated Manufacturing Systems**, v. 8, n. 2, p. 79-89, 1997.

76 PACIFIC EDGE SOFTWARE. **Project Management XML Schema**. Disponível em: <http://www.projectoffice.com/xml/xml.asp> Acesso em: 23 abr. 2003. Definição do padrão XML para dados de projeto.

77 PALMER, J. D.; FIELDS, Computer-Suported Cooperative Work. **IEEE Computer**, p. 15-17, May 1994.

78 PARKER, G. M. **Cross-Functional Teams - Working with Allies, Enemies, and Other Strangers**. San Francisco, CA: Jossey-Bass Publishers, 1994.

79 PATTERSON, M. L. **Accelerating Innovation**. New York, NY: Van Nostrand Reinhold, 1993.

80 PATTERSON, M. L. Minicurso. In: DESENVOLVIMENTO RÁPIDO DE PRODUTOS TECNOLÓGICOS: Florianópolis. **Product Innovation: Leadership Principles for Executives**. Florianópolis: Fundação Certi/Innovation Results International, 1998.

81 PDM IMPLEMENTOR FORUM. **Usage Guide for the STEP PDM Schema, Release 4.3**. Disponível em <http://www.pdm-if.org/> Acesso em: 23 abr. 2003.

82 PRASAD, B. **Concurrent Engineering Fundamentals - Volume I**. Upper Saddle River, NJ: Prentice Hall, 1996a.

83 PRASAD, B. **Concurrent Engineering Fundamentals - Volume II**. Upper Saddle River, NJ: Prentice Hall, 1996b.

84 RAFII, F.; PERKINS, S. Internationalizing Software with Concurrent Engineering. **IEEE Software**, p. 39-46, Sept. 1995.

85 ROMAN, E.; AMBLER, S.; JEWELL, T. **Mastering Enterprise JavaBeans- Second Edition**. New York, NY: John Wiley & Sons, 2002.

86 ROSCHE, Phil. **General Information and Techniques for Improving STEP Translation Success**. Disponível em: <http://www.cax-if.org/> Acesso em: 15 mar. 2003.

87 ROY, U.; USHER, J. M.; PARSAEI, H. R (Ed.). **Simultaneous Engineering: Methodologies and Applications**. Amsterdã: Gordon and Breach Science Publishers, 1999.

88 SCHIMMECK, T. Agents on the Web. **New World**. 2/2002. p. 39-41.

89 SILVA, E. L. da; MENEZES, E. M. **Metodologia da Pesquisa e Elaboração de Dissertação - 3. ed. rev. atual**. Florianópolis: Laboratório de Ensino a Distância da UFSC, 2001.

90 SMITH, P. G.; REINERTSEN, D. G. **Developing Products in Half the Time**. New York, NY: Van Nostrand Reinhold, 1991.

91 SMITH, P., G. Cross-Functional Design Teams. **ASM Handbook**, v. 20, 1997a.

92 SMITH, R. P. The Historical Roots of Concurrent Engineering Fundamentals. **IEEE Transactions on Engineering Management**, v. 44, n. 1, p. 67-78, Feb. 1997b.

93 SMITH, R. P.; EPPINGER, S. D. A Predictive Model of Sequential Iteration in Engineering Design. **Management Science**, v. 43, n. 8, p. 1104-1120, Aug. 1997.

94 SMITH, R. P.; EPPINGER, S. D. Deciding between Sequential and Concurrent Tasks in Engineering Design. **Concurrent Engineering: Research and Applications**, v. 6, n. 1, p. 15-25, mar. 1998.

95 SMITH, R. P.; EPPINGER, S. D. Identifying Controlling Features of Engineering Design Iteration. **Management Science**, v. 43, n. 3, p. 276-292, Mar. 1997.

96 SMITH, R. P.; MORROW, J. A. Product development process modeling. **Design Studies**, v. 20, p. 237-261, 1999.

97 SMITH, R. P.; TJANDRA, P. Experimental Observation in Engineering Design. **Research in Engineering Design**, v. 10, p. 107-117, 1998.

98 SOCIETY OF CONCURRENT PRODUCT DEVELOPMENT (SCPD). **SCPD Body of Knowledge**. Disponível em: <http://www.scpd.org> Acesso em: 23 abr. 2003.

99 STADSZISZ, P. C. **Projeto de Software Usando a UML Versão 2002**. Curitiba: DAINF/DIRCT/CEFET-PR, 2002.

100 SUSMAN, G.; DEAN J. Development of a model for predicting design for manufacturability effectiveness. In: SUSMAN, G. (Ed.). **Integrating Design and Manufacturing for Competitive Advantage**. New York, NY: Oxford, p. 207-227, 1992.

101 TANG, N. K. H.; JONES, O.; FORRESTER, P. L. Organizational growth demands concurrent engineering. **Integrated Manufacturing Systems**, v. 8, n. 1, p 29-34, 1997.

102 THRONICKE, W.; SCHOLZ, R. LIP - A Collaborative Multi-Platform Workflow Environment. In: 5TH EUROPEAN CONCURRENT ENGINEERING CONFERENCE 1998: Erlangen. **Concurrent Engineering: The Way Forward**. Delft: p. 99-103, 1998.

103 TIBBITTS, Capt. B.; KEANE Jr., R. G. Making Design Everybody's Job. **Naval Engineers Journal**, p. 283-301, May 1995.

104 WHEELWRIGHT, S. C.; CLARK, K. B. **Revolutionizing Product Development**. New York, NY: The Free Press, 1992.

105 WINNER, R. I. **The role of concurrent engineering in weapons system acquisition - IDA Report R-338**. Alexandra, VA: Institute of Defense Analysis, 1988. Relatório técnico.

106 WOMACK, J. P.; JONES, D. T.; ROOS, D. **A Máquina que Mudou o Mundo**. São Paulo: Editora Campus, 1992.

APÊNDICES

APÊNDICE 1 - MODELO DE QUESTIONÁRIO

Nome do entrevistado:

Função no processo a ser modelado:

1. Nome da atividade:
2. Como poderia ser a descrição dessa atividade?
3. Qual o procedimento durante a atividade?
4. Quais são as condições de início?
5. Qual é a atividade anterior a ela?
6. Qual é a condição para que ela termine e se prossiga para a atividade seguinte?
7. Quais são os insumos (itens necessários para que seja iniciada)?
8. Qual é o produto dela?
9. Qual é a próxima etapa?
10. Qual a condição para se suspender a atividade?
11. Qual é a condição de finalização da atividade?
12. Quais são as restrições para a atividade?
13. Existe notificação para essa atividade? Como ela é feita e quem é notificado?
14. Essa atividade precisa ser documentada? Com que nível de detalhamento?
15. Qual a duração mínima e máxima da atividade?
16. Que referências são necessárias? Como elas são conseguidas(contatos, fontes de pesquisa, etc.)?
17. Que programa, outros materiais ou infra-estrutura são necessários para a realização dessa atividade?
18. Qual o perfil da pessoa a executar essa atividade (conhecimento necessário)?

APÊNDICE 2 - SCRIPT SQL DE CRIAÇÃO DO BANCO DE DADOS POSTGRESQL

```
CREATE TABLE EMPRESA (
       empresa_cnpj         varchar(300) NOT NULL,
       empresa_nome         varchar(100) NOT NULL,
       empresa_url_intranet varchar(100) NOT NULL,
       empresa_endereco     varchar(200) NOT NULL,
       PRIMARY KEY (empresa_cnpj)
);


CREATE TABLE SETOR (
       setor_id             int NOT NULL,
       empresa_cnpj         varchar(300) NOT NULL,
       setor_nome           varchar(100) NOT NULL,
       setor_fax            varchar(300) NOT NULL,
       PRIMARY KEY (setor_id, empresa_cnpj),
       FOREIGN KEY (empresa_cnpj)
                            REFERENCES EMPRESA
);


CREATE TABLE COLABORADOR (
       colaborador_id       int NOT NULL,
       setor_id             int NOT NULL,
       empresa_cnpj         varchar(300) NOT NULL,
       disponibilidade      float NOT NULL,
       colaborador_nome     varchar(100) NOT NULL,
       colaborador_funcao   varchar(50) NOT NULL,
       colaborador_custo    float NOT NULL,
       colaborador_telefone varchar(300) NOT NULL,
       colaborador_email    varchar(50) NOT NULL,
       colaborador_celular  varchar(300) NOT NULL,
       colaborador_login    varchar(30) NOT NULL,
       colaborador_senha    varchar(30) NOT NULL,
       PRIMARY KEY (colaborador_id, setor_id, empresa_cnpj),
       FOREIGN KEY (setor_id, empresa_cnpj)
                            REFERENCES SETOR
);


CREATE TABLE ARQUIVO_VISUALIZACAO_TIPO (
       arquivo_visualizacao_tipo_id int NOT NULL,
       arquivo_visualizacao_tipo varchar(300) NOT NULL,
       PRIMARY KEY (arquivo_visualizacao_tipo_id)
);


CREATE TABLE ARQUIVO_TIPO (
       arquivo_tipo_id      int NOT NULL,
       arquivo_tipo         varchar(300) NOT NULL,
       PRIMARY KEY (arquivo_tipo_id)
);


CREATE TABLE PROJETO (
       projeto_id           int NOT NULL,
       projeto_nome         varchar(100) NOT NULL,
       projeto_data_inicio  date NULL,
       projeto_data_fim     date NULL,
       projeto_release      int NULL,
       projeto_status       int NULL,
       projeto_marco        int NULL,
       projeto_descricao    varchar(300) NOT NULL,
       projeto_id_cenario   int NOT NULL,
       projeto_workflow     varchar(100) NOT NULL,
       PRIMARY KEY (projeto_id)
);


CREATE TABLE ELEMENTO (
       elemento_id          int NOT NULL,
       projeto_id           int NOT NULL,
       elemento_nome        varchar(100) NOT NULL,
```

```
       elemento_descricao   varchar(300) NOT NULL,
       PRIMARY KEY (elemento_id, projeto_id),
       FOREIGN KEY (projeto_id)
                            REFERENCES PROJETO
);


CREATE TABLE ARQUIVO (
       arquivo_id           int NOT NULL,
       elemento_id          int NOT NULL,
       projeto_id           int NOT NULL,
       arquivo_ultima_versao int NOT NULL,
       PRIMARY KEY (arquivo_id, elemento_id, projeto_id),
       FOREIGN KEY (elemento_id, projeto_id)
                            REFERENCES ELEMENTO
);


CREATE TABLE ARQUIVO_HISTORICO (
       arquivo_historico_id int NOT NULL,
       arquivo_versao       int NOT NULL,
       arquivo_id           int NOT NULL,
       elemento_id          int NOT NULL,
       projeto_id           int NOT NULL,
       arquivo_tipo_id      int NOT NULL,
       arquivo_visualizacao_tipo_id int NOT NULL,
       colaborador_id       int NOT NULL,
       setor_id             int NOT NULL,
       empresa_cnpj         varchar(300) NOT NULL,
       arquivo_fisico       varchar(300) NOT NULL,
       arquivo_visualizacao_fisico varchar(300) NOT NULL,
       arquivo_nome         varchar(300) NOT NULL,
       arquivo_descricao    varchar(300) NOT NULL,
       arquivo_data         date NOT NULL,
       arquivo_tamanho      float NOT NULL,
       arquivo_situacao     varchar(10) NOT NULL,
        PRIMARY KEY (arquivo_historico_id, arquivo_versao, arquivo_id,
              elemento_id, projeto_id, arquivo_tipo_id,
              arquivo_visualizacao_tipo_id, colaborador_id, setor_id,
              empresa_cnpj),
       FOREIGN KEY (colaborador_id, setor_id, empresa_cnpj)
                            REFERENCES COLABORADOR,
       FOREIGN KEY (arquivo_visualizacao_tipo_id)
                            REFERENCES ARQUIVO_VISUALIZACAO_TIPO,
       FOREIGN KEY (arquivo_tipo_id)
                            REFERENCES ARQUIVO_TIPO,
       FOREIGN KEY (arquivo_id, elemento_id, projeto_id)
                            REFERENCES ARQUIVO
);


CREATE TABLE MARCO (
       marco_id             int NOT NULL,
       projeto_id           int NOT NULL,
       marco_nome           varchar(100) NOT NULL,
       marco_descricao      varchar(300) NOT NULL,
       PRIMARY KEY (marco_id, projeto_id),
       FOREIGN KEY (projeto_id)
                            REFERENCES PROJETO
);


CREATE TABLE STATUS (
       status_id            int NOT NULL,
       projeto_id           int NOT NULL,
       status_nome          varchar(100) NOT NULL,
       status_descricao     varchar(300) NOT NULL,
       PRIMARY KEY (status_id, projeto_id),
       FOREIGN KEY (projeto_id)
                            REFERENCES PROJETO
);


CREATE TABLE TAREFA (
       tarefa_id            int NOT NULL,
       marco_id             int NOT NULL,
```

```
	projeto_id		int NOT NULL,
	status_id		int NOT NULL,
	tarefa_mae_id		int NULL,
	duracao_prevista	float NULL,
	data_inicio_prevista	date NULL,
	data_fim_prevista	date NULL,
	descricao		varchar(300) NULL,
	tarefa_nome		varchar(100) NULL,
	composta		int NULL,
	PRIMARY KEY (tarefa_id, marco_id, projeto_id, status_id),
	FOREIGN KEY (marco_id, projeto_id)
				REFERENCES MARCO,
	FOREIGN KEY (tarefa_id, marco_id, projeto_id, status_id)
				REFERENCES TAREFA,
	FOREIGN KEY (status_id, projeto_id)
				REFERENCES STATUS
);


CREATE TABLE EVENTO (
	evento_id		int NOT NULL,
	projeto_id		int NOT NULL,
	colaborador_id		int NOT NULL,
	setor_id		int NOT NULL,
	empresa_cnpj		varchar(300) NOT NULL,
	tipo_evento		varchar(300) NOT NULL,
	data_evento		date NOT NULL,
	arquivo_versao		int NOT NULL,
	arquivo_id		int NOT NULL,
	elemento_id		int NOT NULL,
	arquivo_historico_id	int NOT NULL,
	arquivo_tipo_id		int NOT NULL,
	arquivo_visualizacao_tipo_id int NOT NULL,
	marco_id		int NOT NULL,
	tarefa_id		int NOT NULL,
	status_id		int NOT NULL,
	PRIMARY KEY (evento_id, projeto_id),
	FOREIGN KEY (arquivo_historico_id, arquivo_versao, arquivo_id,
		elemento_id, projeto_id, arquivo_tipo_id,
		arquivo_visualizacao_tipo_id, colaborador_id, setor_id,
		empresa_cnpj)
				REFERENCES ARQUIVO_HISTORICO,
	FOREIGN KEY (projeto_id)
				REFERENCES PROJETO,
	FOREIGN KEY (tarefa_id, marco_id, projeto_id, status_id)
				REFERENCES TAREFA
);


CREATE TABLE REUNIAO (
	reuniao_id		int NOT NULL,
	evento_id		int NULL,
	projeto_id		int NULL,
	data_horario		date NULL,
	duracao			float NULL,
	PRIMARY KEY (reuniao_id),
	FOREIGN KEY (evento_id, projeto_id)
				REFERENCES EVENTO
);


CREATE TABLE ASSUNTO (
	assunto_id		int NOT NULL,
	reuniao_id		int NOT NULL,
	descricao		varchar(300) NULL,
	responsavel_id		int NULL,
	data_realizacao		date NULL,
	definicao		varchar(300) NOT NULL,
	PRIMARY KEY (assunto_id, reuniao_id),
	FOREIGN KEY (reuniao_id)
				REFERENCES REUNIAO
);


CREATE TABLE DISCUSSAO (
	discussao_id		int NOT NULL,
```

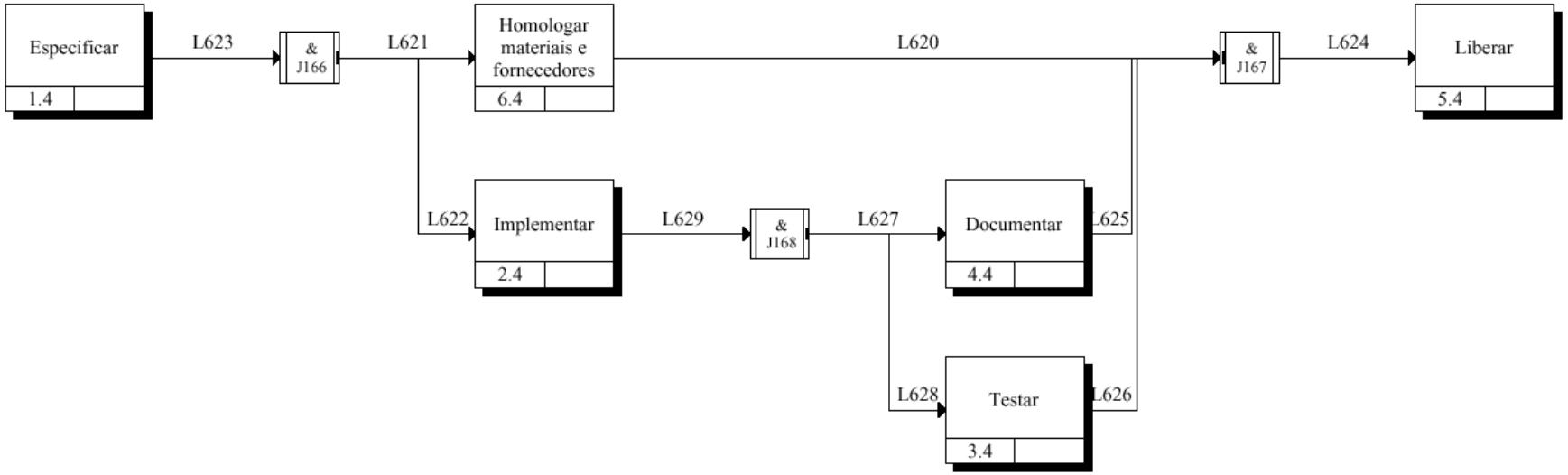
```
	evento_id		int NULL,
	projeto_id		int NULL,
	conteudo		varchar(300) NULL,
	data			date NULL,
	PRIMARY KEY (discussao_id),
	FOREIGN KEY (evento_id, projeto_id)
				REFERENCES EVENTO
);


CREATE TABLE METRICA (
	metrica_id		int NOT NULL,
	projeto_id		int NOT NULL,
	metrica			varchar(5) NOT NULL,
	metrica_valor		float NOT NULL,
	PRIMARY KEY (metrica_id, projeto_id),
	FOREIGN KEY (projeto_id)
				REFERENCES PROJETO
);


CREATE TABLE VAULT (
	vault_id		int NOT NULL,
	url			varchar(300) NOT NULL,
	vault_nome		varchar(300) NOT NULL,
	ftplogin		varchar(300) NOT NULL,
	ftppasswd		varchar(300) NOT NULL,
	PRIMARY KEY (vault_id)
);


CREATE TABLE NIVEL_ABSTRACAO (
	nivel_abstracao		int NOT NULL,
	nivel_abstracao_descricao varchar(300) NOT NULL,
	PRIMARY KEY (nivel_abstracao)
);


CREATE TABLE DEPENDENCIA_ELEMENTO (
	dependencia_elemento_id int NOT NULL,
	elemento_id		int NOT NULL,
	projeto_id		int NOT NULL,
	elemento_referencia_id int NOT NULL,
	PRIMARY KEY (dependencia_elemento_id, elemento_id, projeto_id),
	FOREIGN KEY (elemento_id, projeto_id)
				REFERENCES ELEMENTO
);


CREATE TABLE RAMO_ARVORE (
	ramo_arvore_id		int NOT NULL,
	projeto_id		int NOT NULL,
	vault_id		int NOT NULL,
	ramo_nome		varchar(100) NOT NULL,
	ramo_pai_id		int NOT NULL,
	ramo_descricao		varchar(300) NOT NULL,
	PRIMARY KEY (ramo_arvore_id, projeto_id),
	FOREIGN KEY (vault_id)
				REFERENCES VAULT,
	FOREIGN KEY (ramo_arvore_id, projeto_id)
				REFERENCES RAMO_ARVORE,
	FOREIGN KEY (projeto_id)
				REFERENCES PROJETO
);


CREATE TABLE ELEMENTO_RAMO (
	elemento_ramo_id	int NOT NULL,
	ramo_arvore_id		int NOT NULL,
	elemento_id		int NOT NULL,
	projeto_id		int NOT NULL,
	PRIMARY KEY (elemento_ramo_id, ramo_arvore_id, elemento_id,
		projeto_id),
	FOREIGN KEY (elemento_id, projeto_id)
				REFERENCES ELEMENTO,
	FOREIGN KEY (ramo_arvore_id, projeto_id)
```

```
                            REFERENCES RAMO_ARVORE
);


CREATE TABLE VINCULACAO_TAREFA (
       vinculacao_tarefa_id int NOT NULL,
       tarefa_id            int NOT NULL,
       status_id            int NOT NULL,
       projeto_id           int NOT NULL,
       marco_id             int NOT NULL,
       tipo_vinculacao      int NULL,
       tarefa_referencia_id int NULL,
       PRIMARY KEY (vinculacao_tarefa_id, tarefa_id, status_id,
              projeto_id, marco_id),
       FOREIGN KEY (tarefa_id, marco_id, projeto_id, status_id)
                            REFERENCES TAREFA
);


CREATE TABLE ELEMENTO_TAREFA (
       elemento_tarefa_id   int NOT NULL,
       tarefa_id            int NOT NULL,
       projeto_id           int NOT NULL,
       status_id            int NOT NULL,
       elemento_id          int NOT NULL,
       marco_id             int NOT NULL,
       PRIMARY KEY (elemento_tarefa_id, tarefa_id, projeto_id,
              status_id, elemento_id, marco_id),
       FOREIGN KEY (elemento_id, projeto_id)
                            REFERENCES ELEMENTO,
       FOREIGN KEY (tarefa_id, marco_id, projeto_id, status_id)
                            REFERENCES TAREFA
);


CREATE TABLE HISTORICO_TAREFA (
       historico_tarefa_id  int NOT NULL,
       tarefa_id            int NOT NULL,
       status_id            int NOT NULL,
       projeto_id           int NOT NULL,
       marco_id             int NOT NULL,
       levantamento_data    date NOT NULL,
       porcentagem          float NULL,
       data_inicio_real     date NOT NULL,
       data_fim_real        date NOT NULL,
       duracao_real         float NOT NULL,
       semaforo             varchar(300) NOT NULL,
       PRIMARY KEY (historico_tarefa_id, tarefa_id, status_id,
              projeto_id, marco_id),
       FOREIGN KEY (tarefa_id, marco_id, projeto_id, status_id)
                            REFERENCES TAREFA
);


CREATE TABLE TAREFA_COLABORADOR (
       tarefa_colaborador_id int NOT NULL,
       tarefa_id            int NOT NULL,
       status_id            int NOT NULL,
       colaborador_id       int NOT NULL,
       projeto_id           int NOT NULL,
       marco_id             int NOT NULL,
       setor_id             int NOT NULL,
       empresa_cnpj         varchar(300) NOT NULL,
       PRIMARY KEY (tarefa_colaborador_id, tarefa_id, status_id,
              colaborador_id, projeto_id, marco_id, setor_id,
              empresa_cnpj),
       FOREIGN KEY (colaborador_id, setor_id, empresa_cnpj)
                            REFERENCES COLABORADOR,
       FOREIGN KEY (tarefa_id, marco_id, projeto_id, status_id)
                            REFERENCES TAREFA
);


CREATE TABLE AREA (
       area_id              int NOT NULL,
       area_nome            varchar(100) NOT NULL,
```

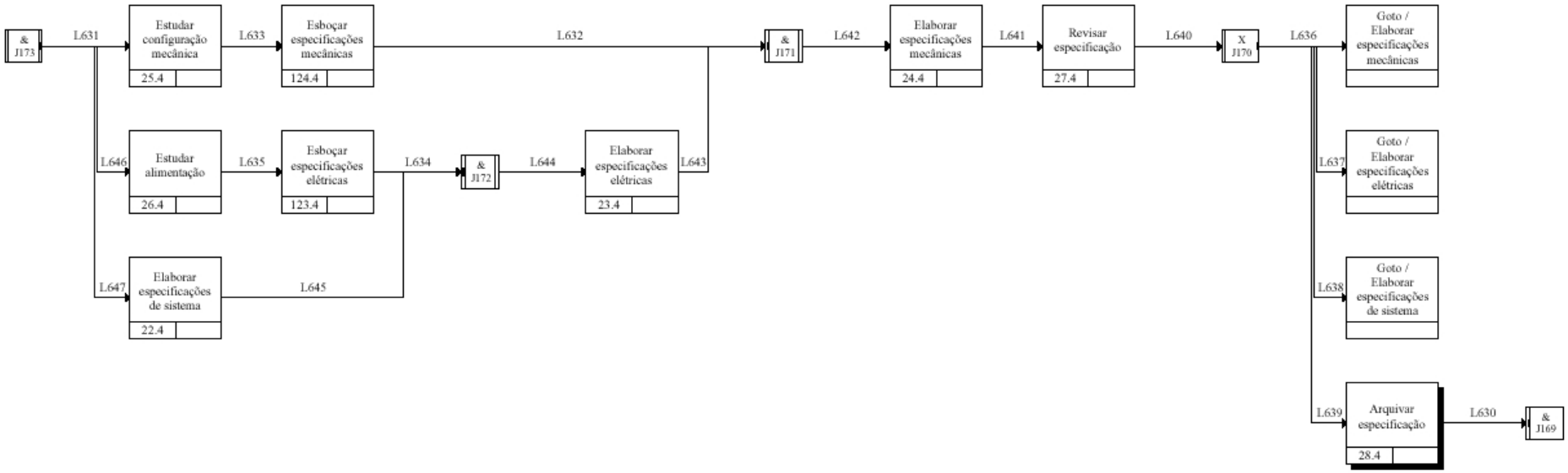
```
	area_descricao	varchar(300) NOT NULL,
	PRIMARY KEY (area_id)
);


CREATE TABLE PROJETO_ABSTRACAO_COLABORADOR (
	papel_colaborador_id int NOT NULL,
	nivel_abstracao	int NOT NULL,
	colaborador_id	int NOT NULL,
	projeto_id	int NOT NULL,
	setor_id	int NOT NULL,
	empresa_cnpj	varchar(300) NOT NULL,
	area_id	int NOT NULL,
	tempo_dedicacao	float NOT NULL,
	PRIMARY KEY (papel_colaborador_id, nivel_abstracao,
		colaborador_id, projeto_id, setor_id, empresa_cnpj,
		area_id),
	FOREIGN KEY (area_id)
				REFERENCES AREA,
	FOREIGN KEY (nivel_abstracao)
				REFERENCES NIVEL_ABSTRACAO,
	FOREIGN KEY (projeto_id)
				REFERENCES PROJETO,
	FOREIGN KEY (colaborador_id, setor_id, empresa_cnpj)
				REFERENCES COLABORADOR
);


CREATE TABLE GRUPO (
	grupo_id	int NOT NULL,
	grupo_descricao	varchar(300) NOT NULL,
	grupo_nome	varchar(100) NOT NULL,
	PRIMARY KEY (grupo_id)
);


CREATE TABLE GRUPO_COLABORADOR (
	grupo_colaborador_id int NOT NULL,
	grupo_id	int NOT NULL,
	colaborador_id	int NOT NULL,
	setor_id	int NOT NULL,
	empresa_cnpj	varchar(300) NOT NULL,
	PRIMARY KEY (grupo_colaborador_id, grupo_id, colaborador_id,
		setor_id, empresa_cnpj),
	FOREIGN KEY (colaborador_id, setor_id, empresa_cnpj)
				REFERENCES COLABORADOR,
	FOREIGN KEY (grupo_id)
				REFERENCES GRUPO
);


CREATE TABLE COMENTARIO (
	comentario_id	int NOT NULL,
	discussao_id	int NOT NULL,
	colaborador_id	int NULL,
	setor_id	int NULL,
	empresa_cnpj	varchar(300) NULL,
	texto	varchar(300) NULL,
	comentario_pai_id	int NULL,
	PRIMARY KEY (comentario_id, discussao_id),
	FOREIGN KEY (colaborador_id, setor_id, empresa_cnpj)
				REFERENCES COLABORADOR,
	FOREIGN KEY (comentario_id, discussao_id)
				REFERENCES COMENTARIO,
	FOREIGN KEY (discussao_id)
				REFERENCES DISCUSSAO
);


CREATE TABLE CONSELHO_REUNIAO (
	conselho_reuniao_id	int NOT NULL,
	reuniao_id	int NOT NULL,
	setor_id	int NULL,
	empresa_cnpj	varchar(300) NULL,
	colaborador_id	int NULL,
	PRIMARY KEY (conselho_reuniao_id, reuniao_id),
```

```
	FOREIGN KEY (colaborador_id, setor_id, empresa_cnpj)
				REFERENCES COLABORADOR,
	FOREIGN KEY (reuniao_id)
				REFERENCES REUNIAO
);


CREATE TABLE SESSAO_COLABORATIVA (
	sessao_colaborativa_id int NOT NULL,
	evento_id		int NULL,
	projeto_id		int NULL,
	data			date NULL,
	descricao		varchar(300) NULL,
	ativa			int NULL,
	duracao			float NULL,
	PRIMARY KEY (sessao_colaborativa_id),
	FOREIGN KEY (evento_id, projeto_id)
				REFERENCES EVENTO
);


CREATE TABLE CONSELHO_SESSAO (
	conselho_sessao_id   int NOT NULL,
	sessao_colaborativa_id int NOT NULL,
	colaborador_id		int NOT NULL,
	setor_id		int NULL,
	empresa_cnpj		varchar(300) NULL,
	endereco_ip		varchar(300) NULL,
	sessao_entrada		date NULL,
	sessao_saida		date NULL,
	ativo			int NULL,
	PRIMARY KEY (conselho_sessao_id, sessao_colaborativa_id,
		colaborador_id),
	FOREIGN KEY (colaborador_id, setor_id, empresa_cnpj)
				REFERENCES COLABORADOR,
	FOREIGN KEY (sessao_colaborativa_id)
				REFERENCES SESSAO_COLABORATIVA
);


CREATE TABLE LOG_SESSAO_COLABORATIVA (
	log_sessao_colaborativa_id int NOT NULL,
	sessao_colaborativa_id int NOT NULL,
	texto			varchar(300) NULL,
	espaco			int NULL,
	grafico			bytea NULL,
	x1			float NULL,
	y1			float NULL,
	x2			float NULL,
	y2			float NULL,
	cor			varchar(10) NULL,
	tipo_log		int NULL,
	tab_id			int NULL,
	PRIMARY KEY (log_sessao_colaborativa_id,
		sessao_colaborativa_id),
	FOREIGN KEY (sessao_colaborativa_id)
				REFERENCES SESSAO_COLABORATIVA
);

CREATE TABLE TEMP AS SELECT * FROM EVENTO;
DROP TABLE EVENTO;
CREATE TABLE EVENTO AS SELECT * FROM TEMP;
DROP TABLE TEMP;
```

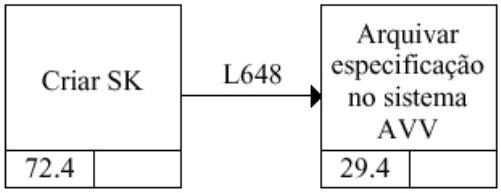
APÊNDICE 3 - PROCESSO DE DESENVOLVIMENTO DE ARMÁRIOS DE TELECOMUNICAÇÕES CAPTURADO PELO MÉTODO IDEF3



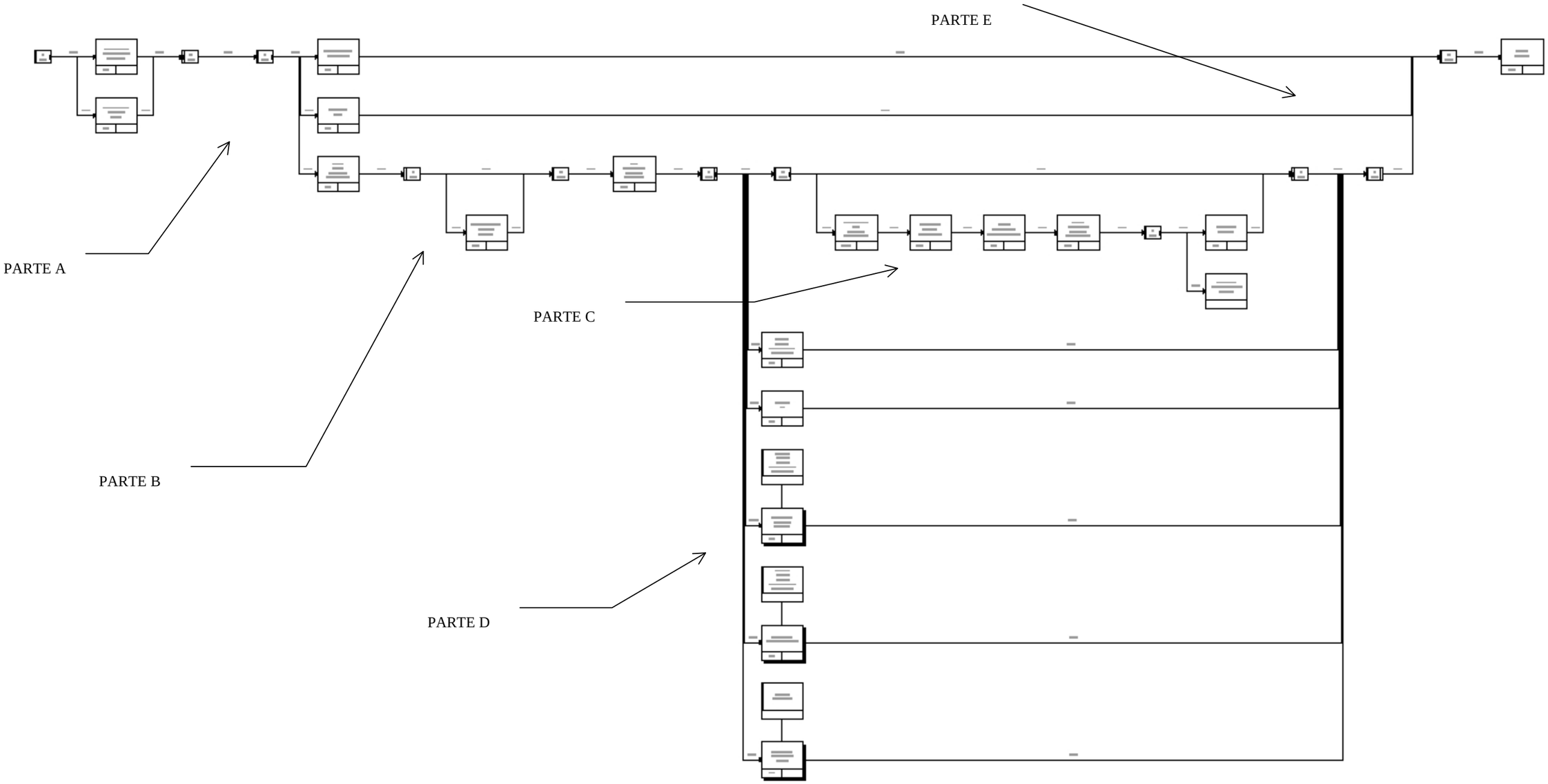
TAREFA COMPOSTA **ESPECIFICAR**



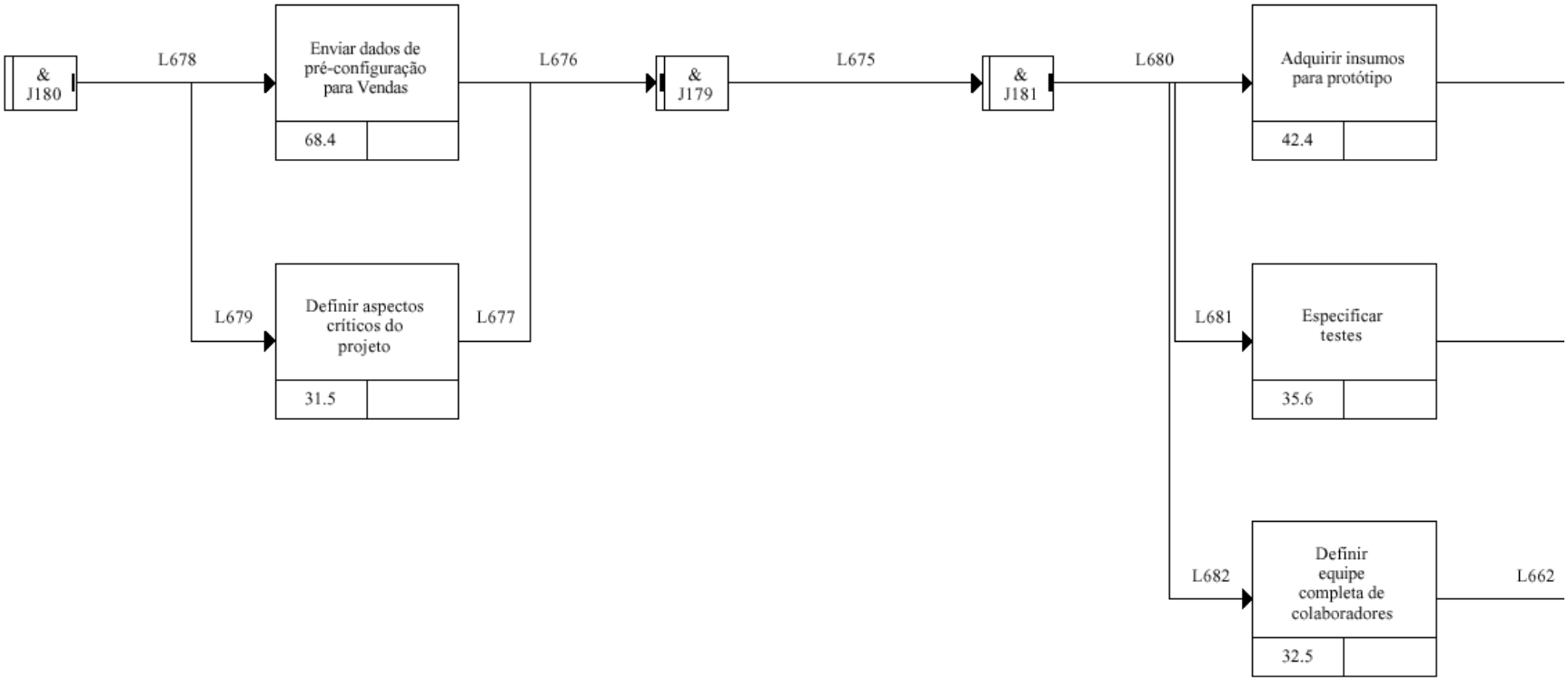
TAREFA COMPOSTA **ARQUIVAR ESPECIFICAÇÃO**



TAREFA COMPOSTA **IMPLEMENTAR**

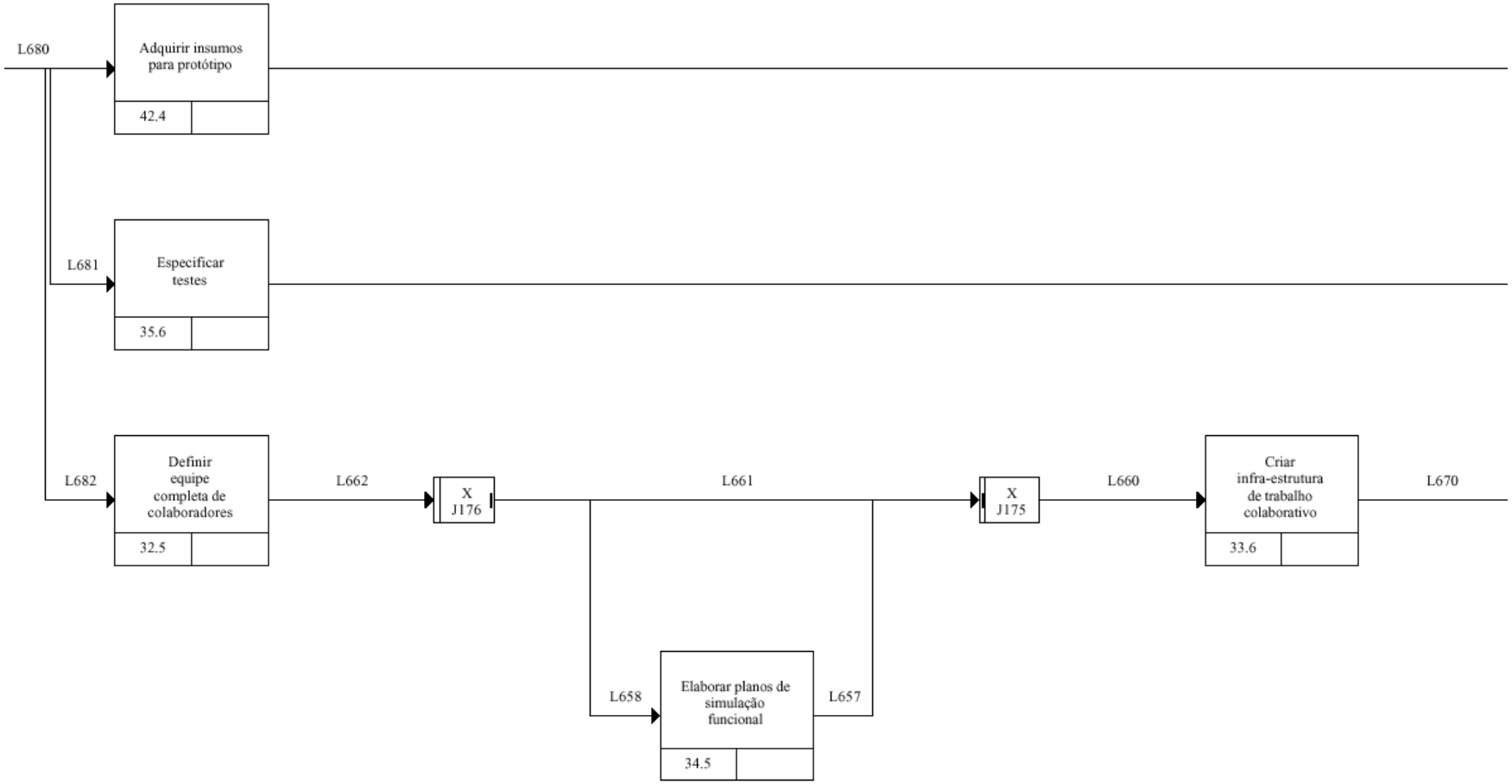


TAREFA COMPOSTA **IMPLEMENTAR (PARTE A)**



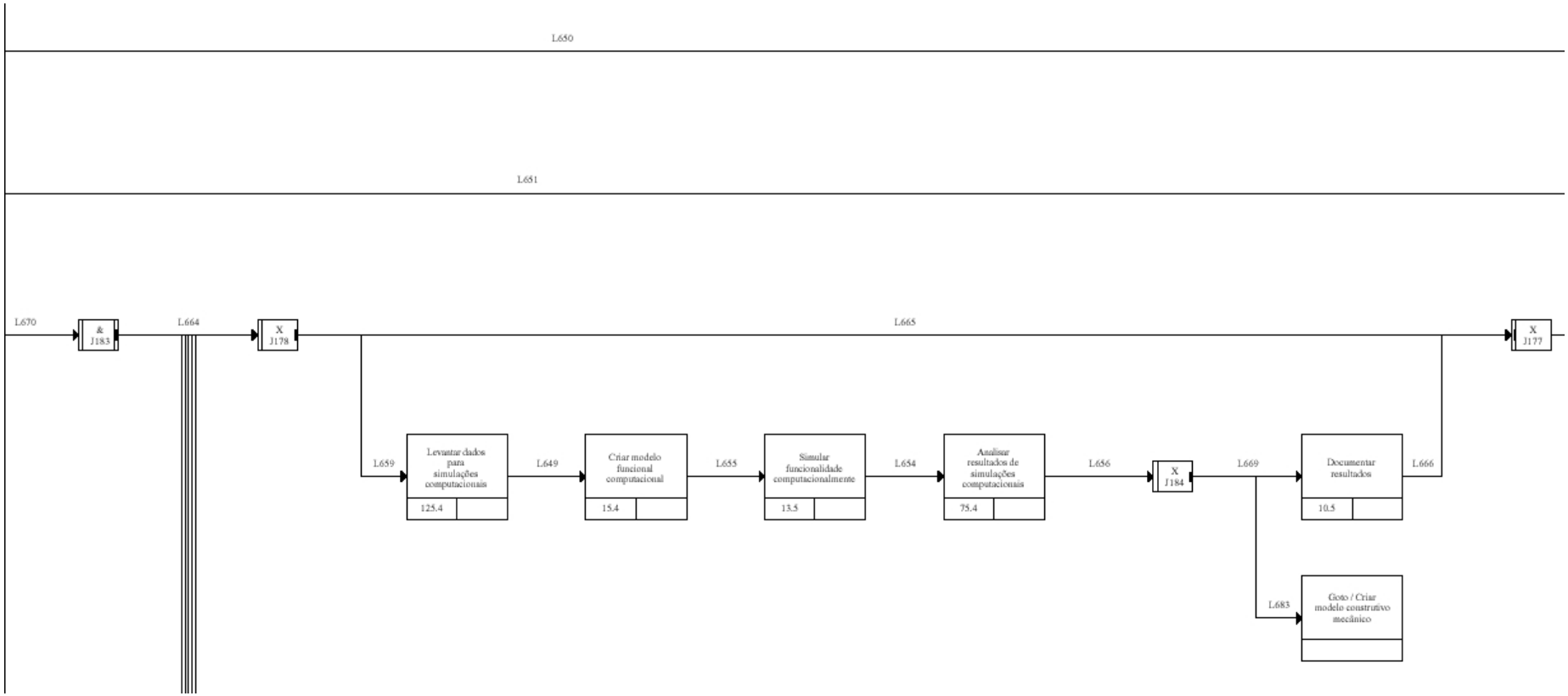
(continua)

TAREFA COMPOSTA **IMPLEMENTAR (PARTE B)**



(continua)

TAREFA COMPOSTA **IMPLEMENTAR (PARTE C)**

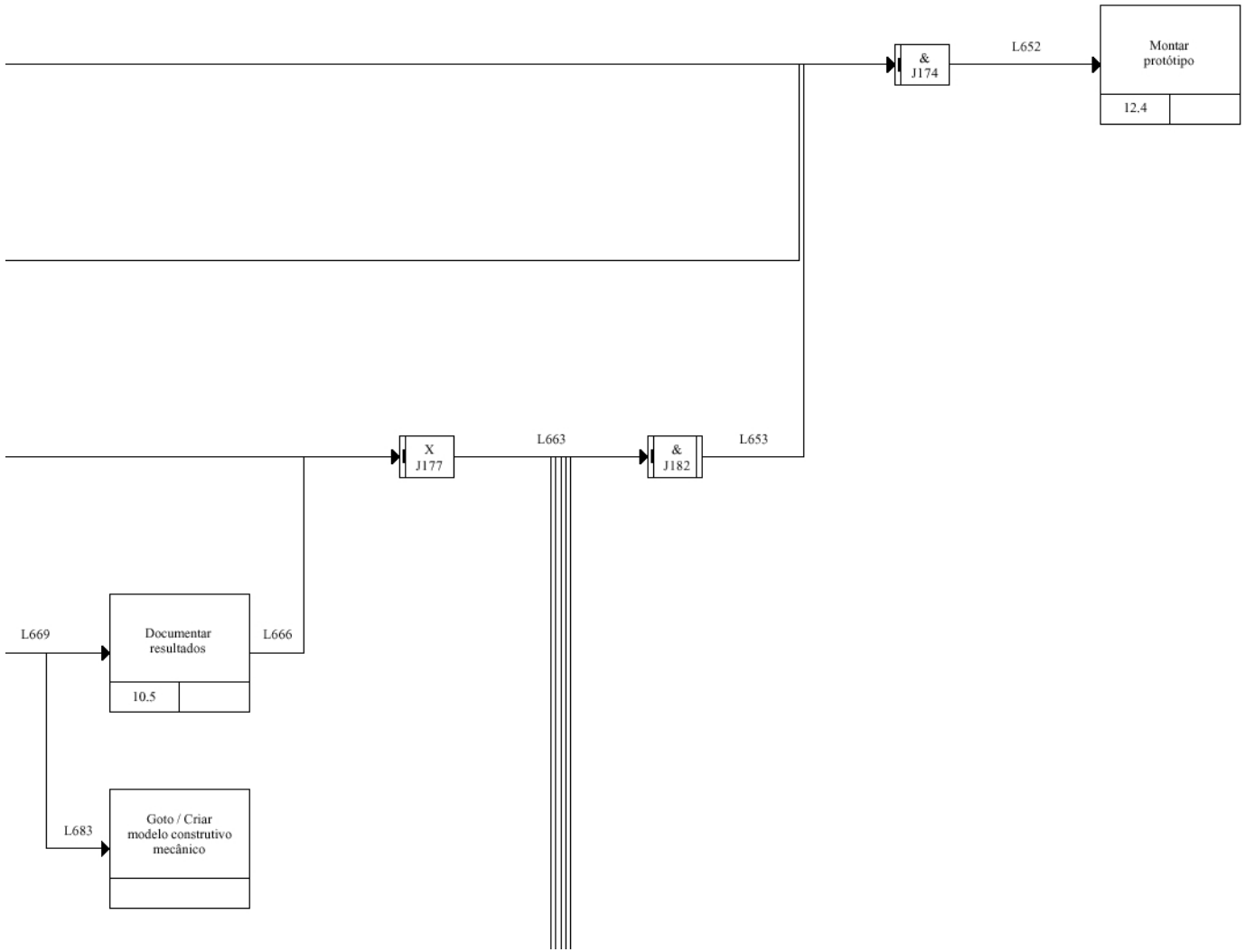


(continua)

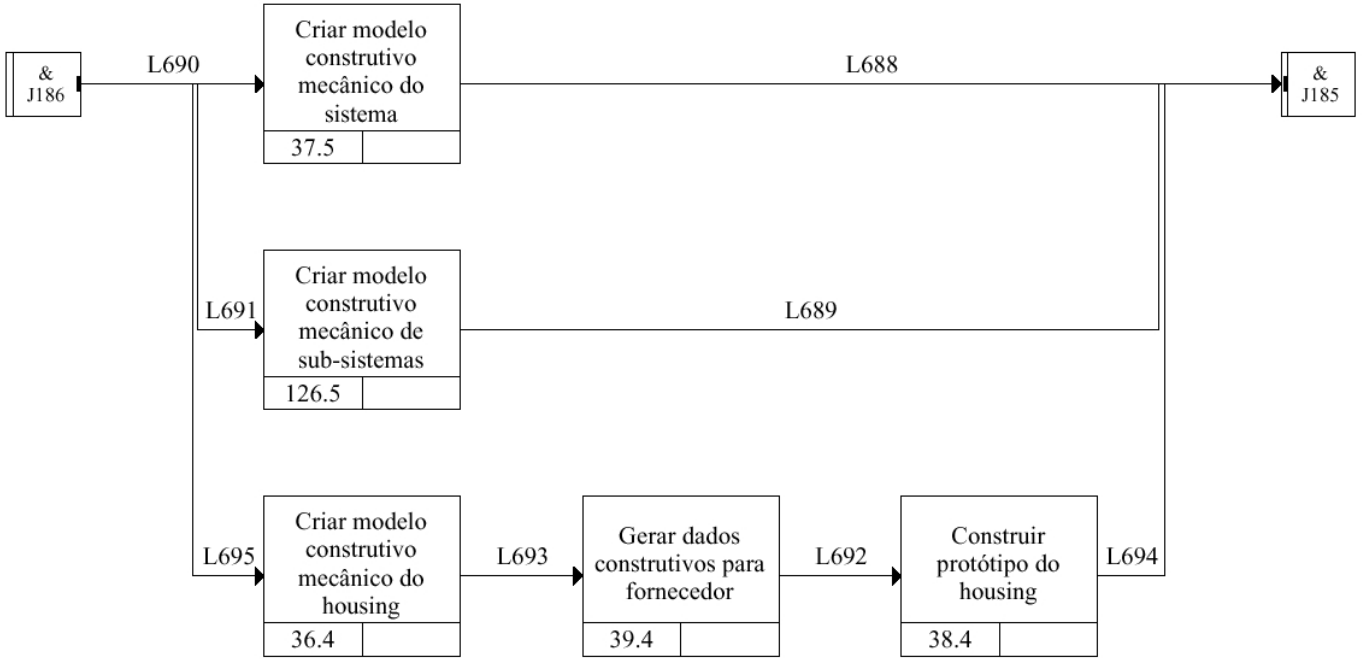
TAREFA COMPOSTA **IMPLEMENTAR (PARTE D)**

(continua)

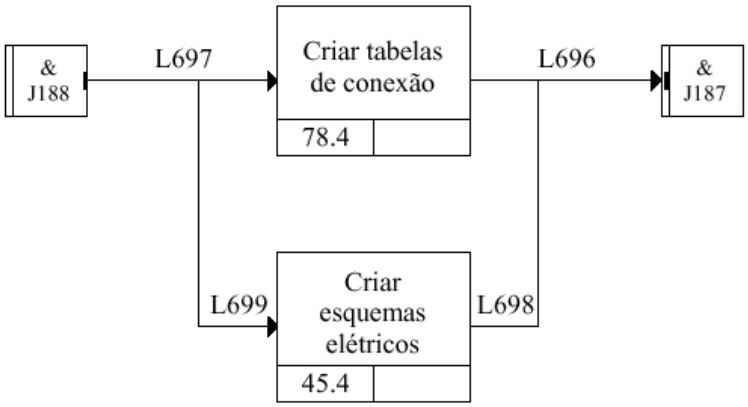
TAREFA COMPOSTA **IMPLEMENTAR (PARTE E)**



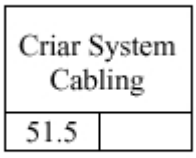
TAREFA COMPOSTA **CRIAR MODELO CONSTRUTIVO MECÂNICO**



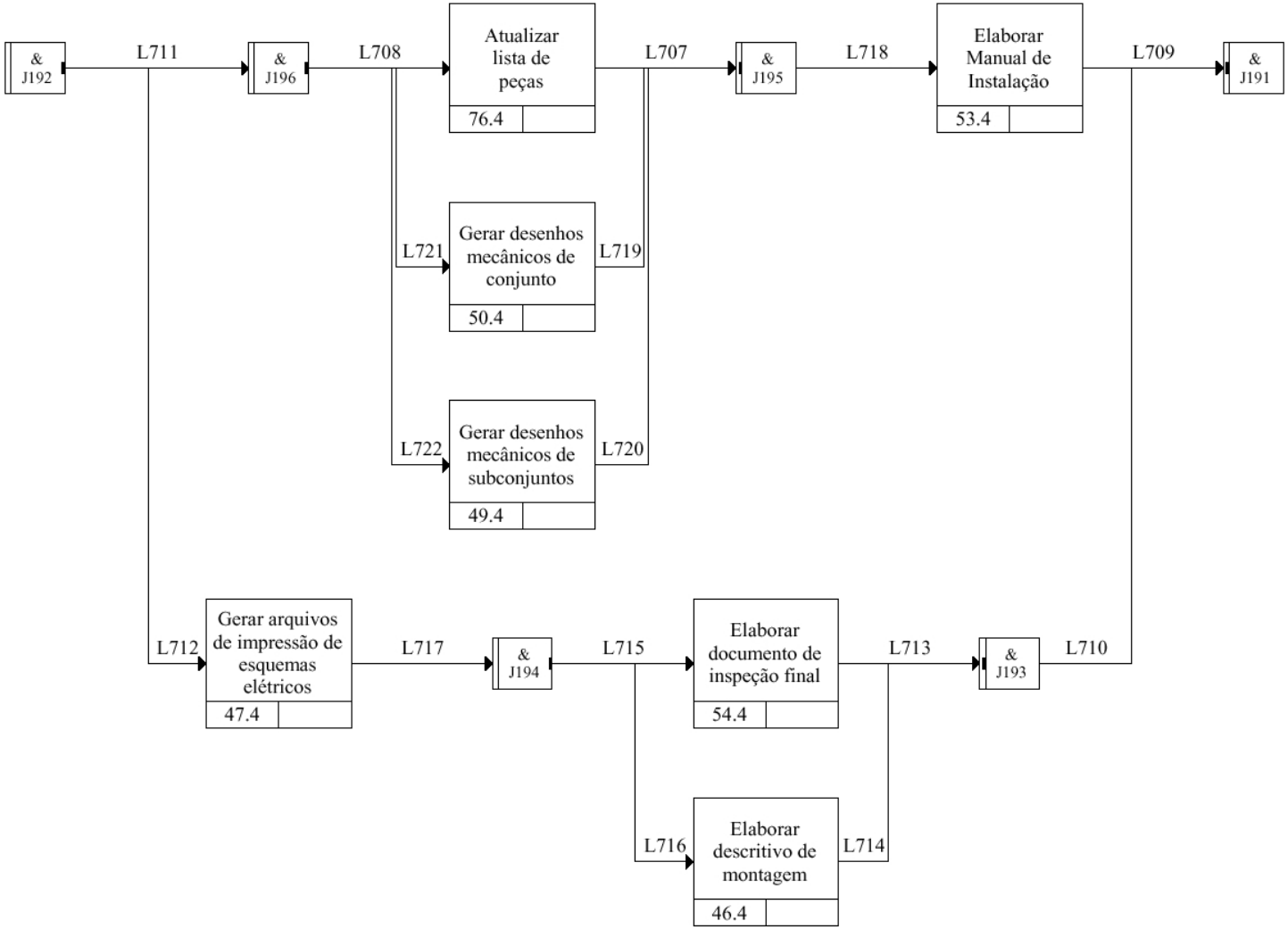
TAREFA COMPOSTA **CRIAR MODELO CONSTRUTIVO ELÉTRICO**



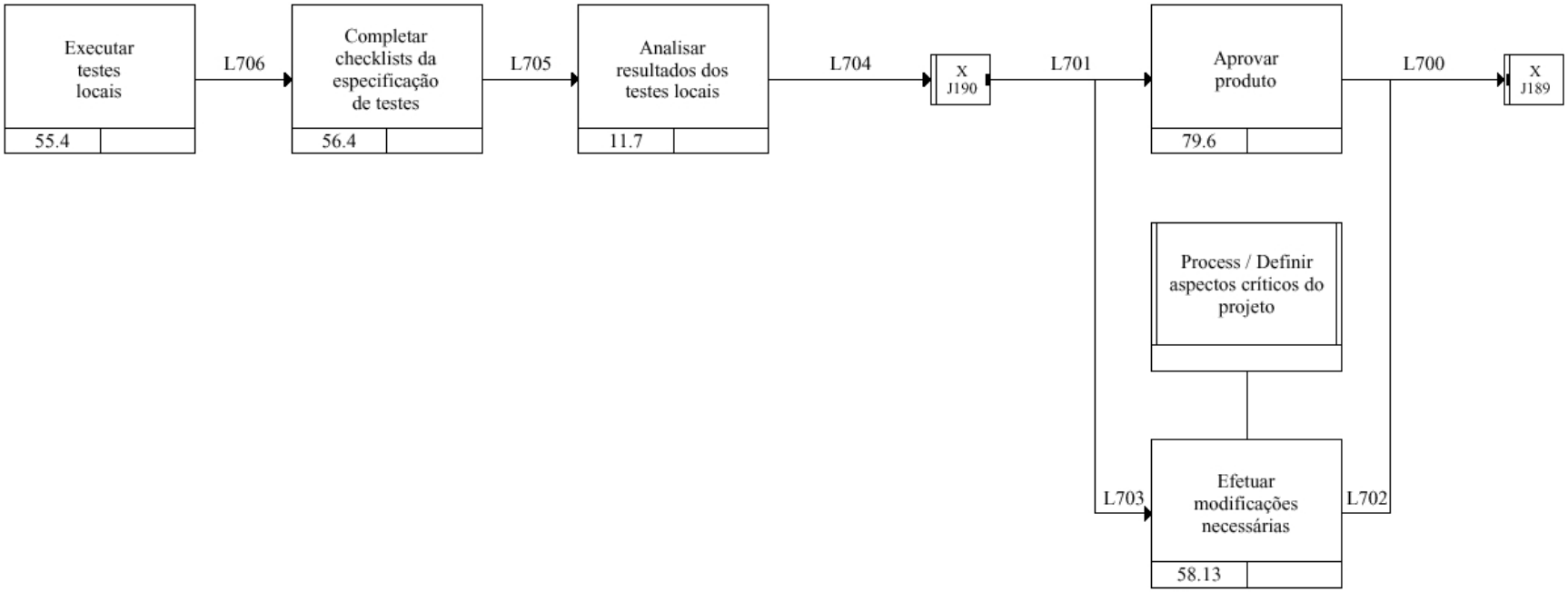
TAREFA COMPOSTA **CRIAR MODELO CONSTRUTIVO DO SISTEMA**



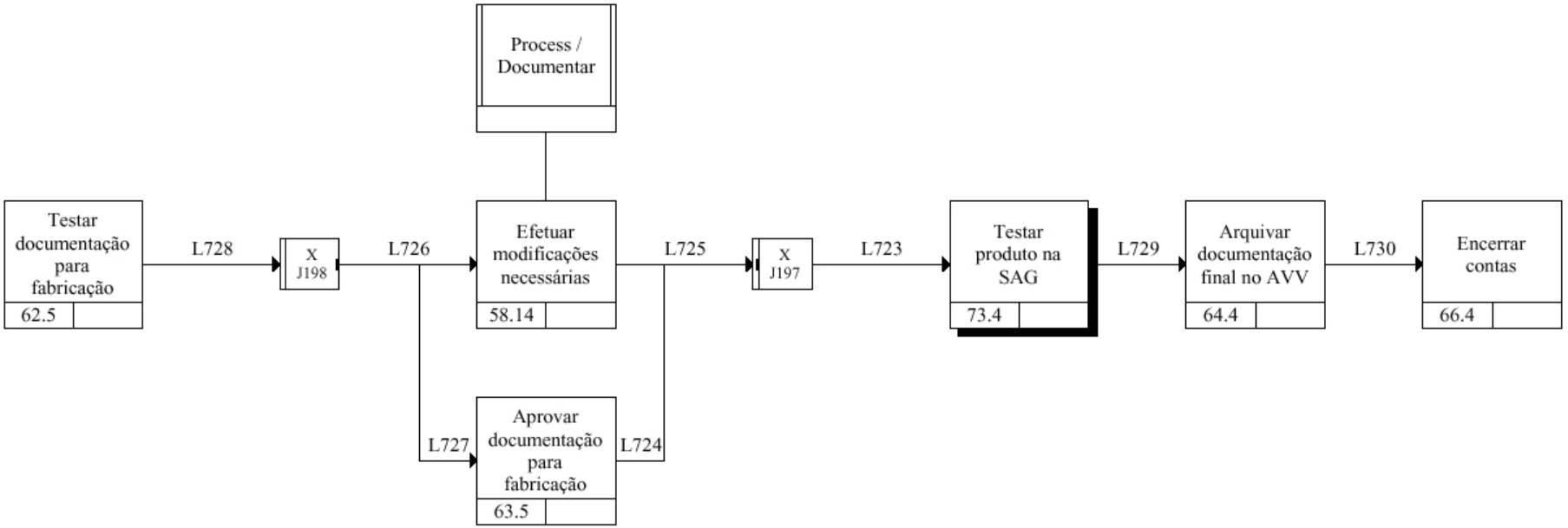
TAREFA COMPOSTA **DOCUMENTAR**



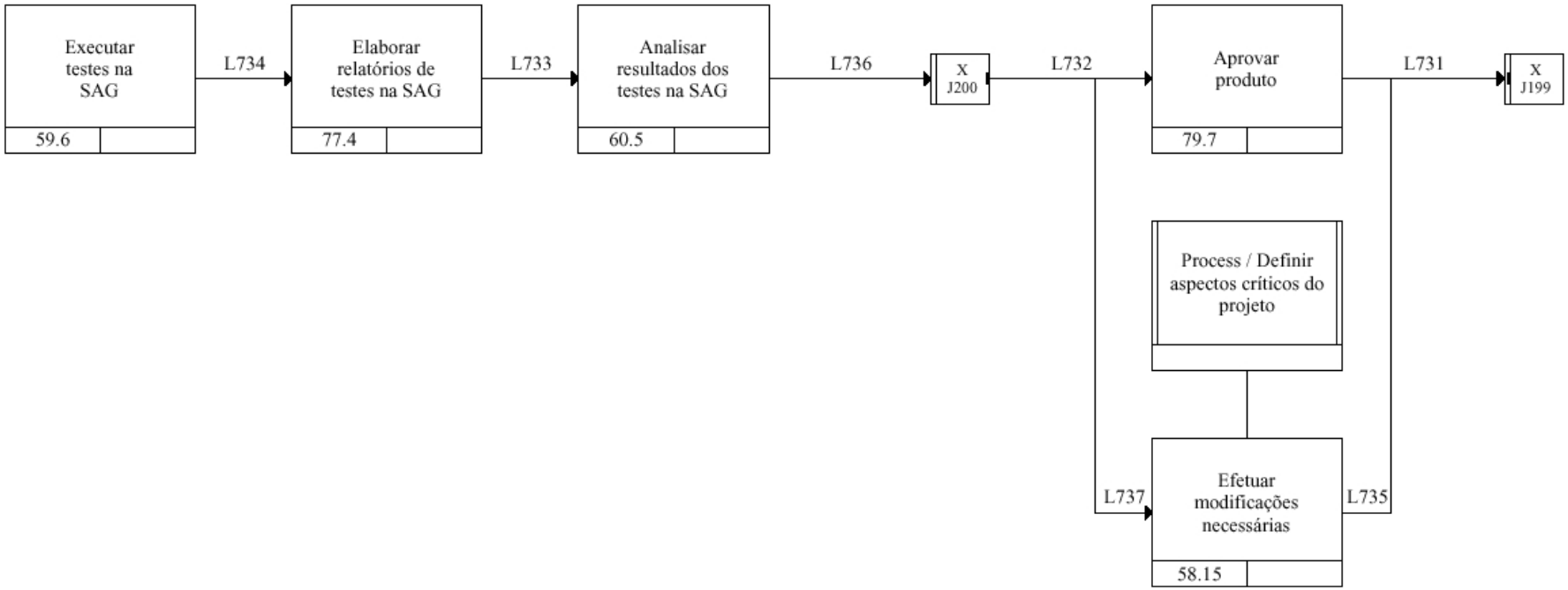
TAREFA COMPOSTA **TESTAR**



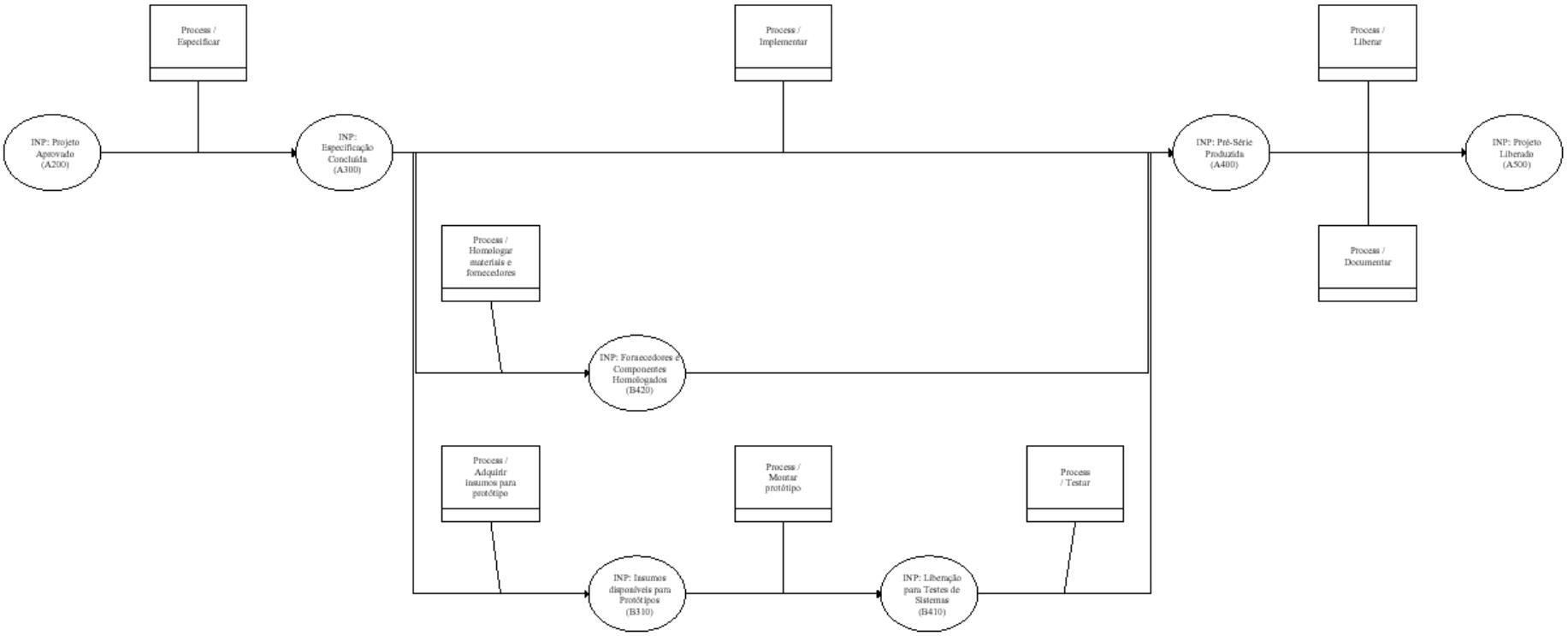
TAREFA COMPOSTA **LIBERAR**



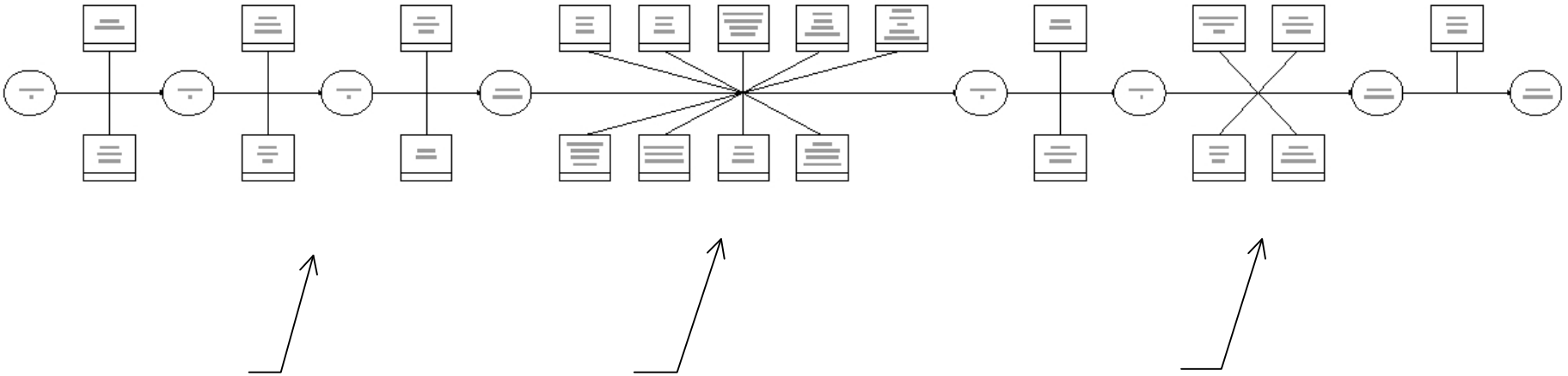
TAREFA COMPOSTA **TESTAR PRODUTO NA SAG**



APÊNDICE 4 - OSTN MARCOS DE PROJETO

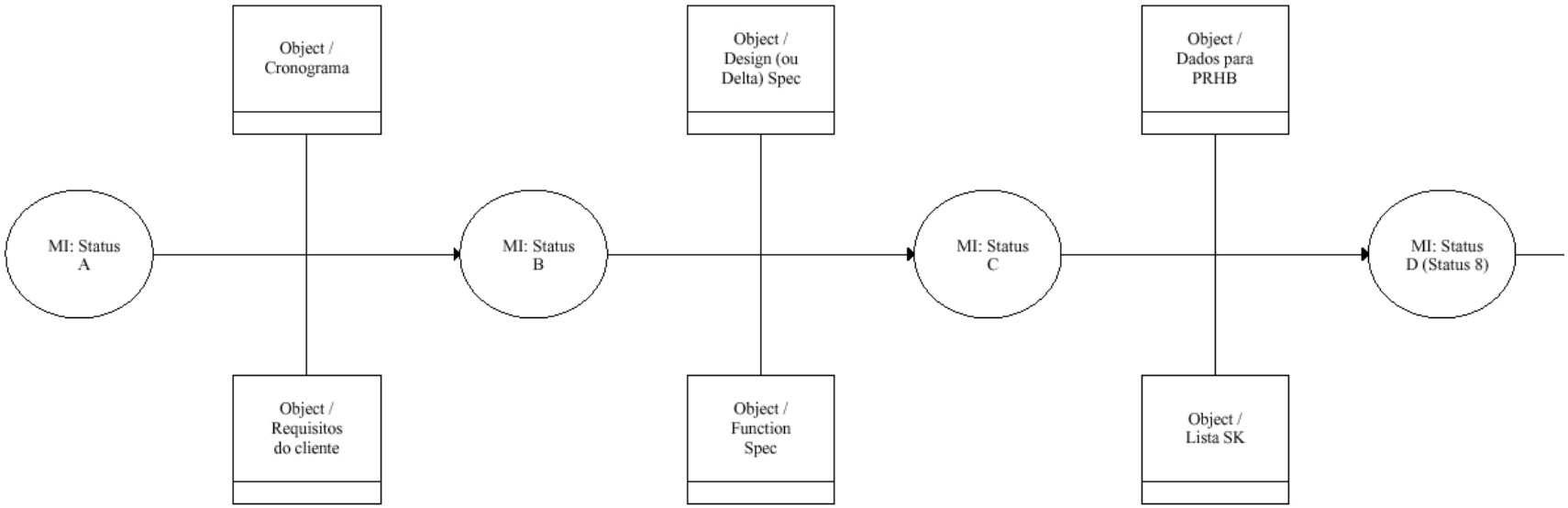


APÊNDICE 5 - OSTN STATUS DA INFORMAÇÃO

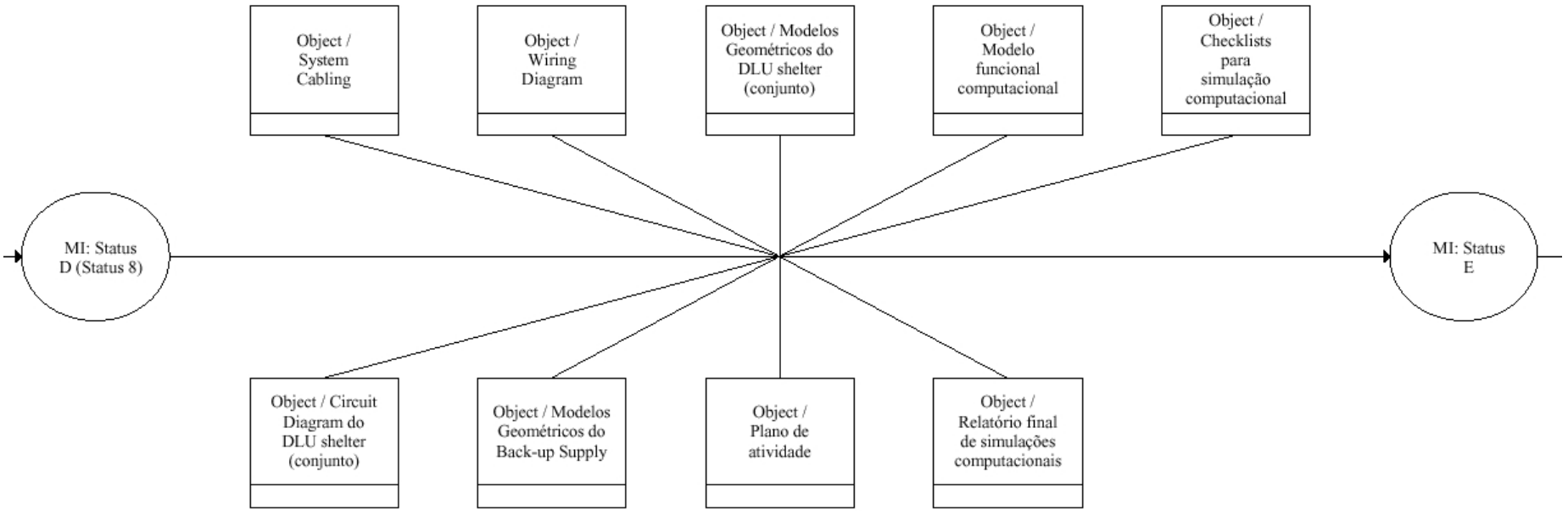


PARTE A PARTE B PARTE C

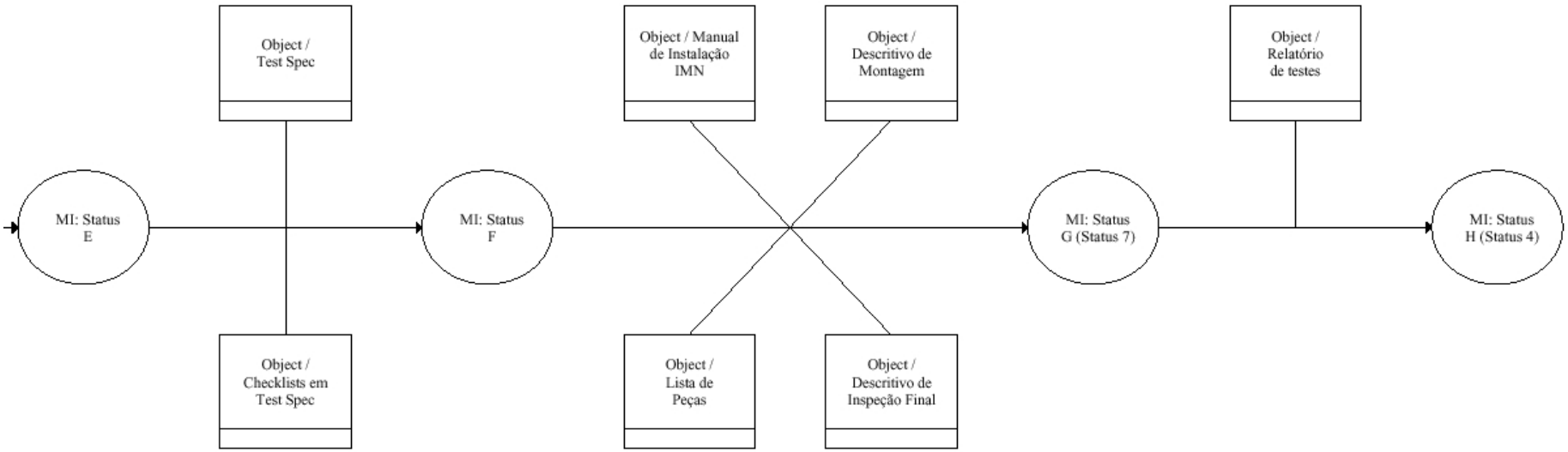
OSTN STATUS DA INFORMAÇÃO (PARTE A)



OSTN STATUS DA INFORMAÇÃO (PARTE B)



OSTN DA INFORMAÇÃO (PARTE C)



APÊNDICE 6 - MATRIZ DE VINCULAÇÃO ENTRE TAREFAS E ELEMENTOS DE INFORMAÇÃO (NÍVEL SUPERIOR)

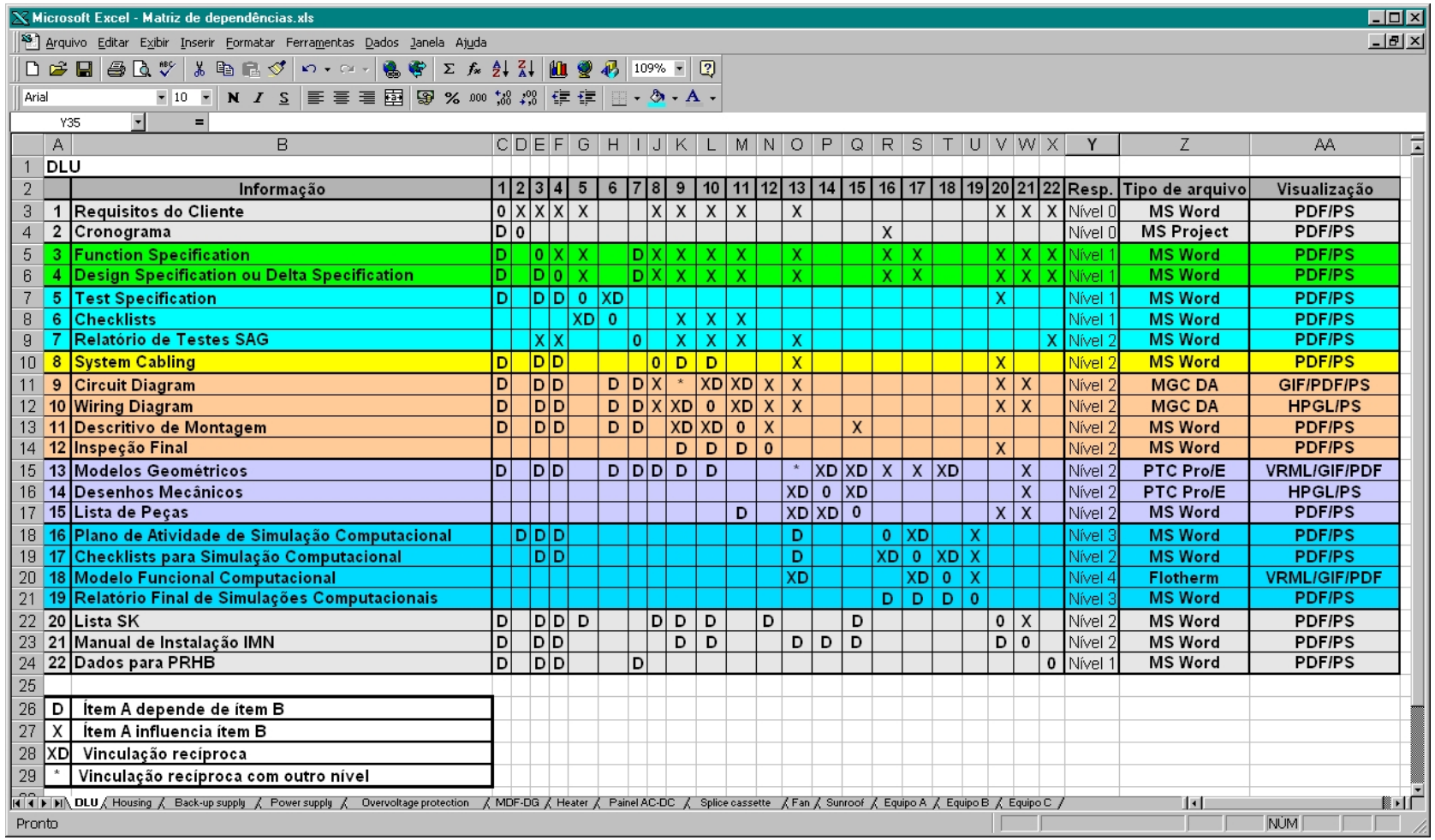
Legend:
E = Entity
L = Location
Q = Queue
S = Resource
T = Transport
I = Logical
? = Conflict
- = Default

Processes	Objects	Aprovação de documentação para fabricação	Aprovação do produto	Arquivamento de dados	Catálogos de fornecedores	Checklists em Test Spec	Checklists para simulação computacional	Circuit Diagram do Back-up Supply	Circuit Diagram do DLU shelter (conjunto)	Circuit Diagram do Heater	Circuit Diagram do MDF-DG	Circuit Diagram do Overvoltage Protection	Circuit Diagram do Painel AC/DC	Circuit Diagram do Power Supply	Criação de contas e grupos nos sistemas	Cronograma	Dados Construtivos do Housing	Dados para PRHB	Descritivo de Inspeção Final	Descritivo de Montagem	Desenhos mecânicos do Back-up Supply	Desenhos mecânicos do DLU shelter (conjunto)	Desenhos mecânicos do Equipo A	Desenhos mecânicos do Equipo B	Desenhos mecânicos do Equipo C	Desenhos mecânicos do Fan	Desenhos mecânicos do Heater	Desenhos mecânicos do MDF-DG	Desenhos mecânicos do Overvoltage Protection	Desenhos mecânicos do Painel AC/DC	Desenhos mecânicos do Power Supply	Design (ou Delta) Spec	Divulgação de resultados de testes	Encerramento de contas	Esboço de especificações elétricas	Esboço de especificações mecânicas	Esquemas elétricos em formato de impressão	Estudo de alimentação
		E	E	E	S	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E
Documentar	L																		X	X		X															X	
Especificar	L																															X						
Homologar materiais e fornecedores	L																																				
Implementar	L								X																													
Liberar	L			X																																	
Testar	L					X																															

Legend:
E = Entity
L = Location
Q = Queue
S = Resource
T = Transport
I = Logical
? = Conflict
- = Default

Processes	Objects	Estudo de configuração mecânica	Formação da equipe de trabalho	Fornecedor de housing	Function Spec	Fábrica	Homologação de materiais e fornecedores	Insumos para protótipo	Lista de Peças	Lista SK	Manual de Instalação IMN	Modelo funcional computacional	Modelo Geométrico do Housing	Modelos Geométricos do Back-up Supply	Modelos Geométricos do DLU shelter (conjunto)	Modelos Geométricos do Equipo A	Modelos Geométricos do Equipo B	Modelos Geométricos do Equipo C	Modelos Geométricos do Fan	Modelos Geométricos do Heater	Modelos Geométricos do MDF-DG	Modelos Geométricos do Overvoltage Protection	Modelos Geométricos do Painel AC/DC	Modelos Geométricos do Power Supply	Modelos Geométricos do Splice Cassette	Modelos Geométricos do Sunroof	NuPES	Plano de atividade	Posto de trabalho Ansys	Posto de trabalho Flotherm	Posto de trabalho Mentor Graphics	Posto de trabalho Office	Posto de trabalho Pro/ENGINEER	Protótipo	Protótipo de housing	Relatório de testes	Relatório de testes sobre documentação	Relatório final de simulações computacionais
		E	E	S	E	S	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	L	E	S	S	S	S	S	E	E	E	E
Documentar	L				X				X		X																											
Especificar	L						X																															
Homologar materiais e fornecedores	L									X					X																							
Implementar	L																																				
Liberar	L					X																															
Testar	L																																				

APÊNDICE 7 - MATRIZ DE DEPENDÊNCIAS DE ARMÁRIOS DE TELECOMUNICAÇÕES



Microsoft Excel - Matriz de dependências.xls

Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Ajuda

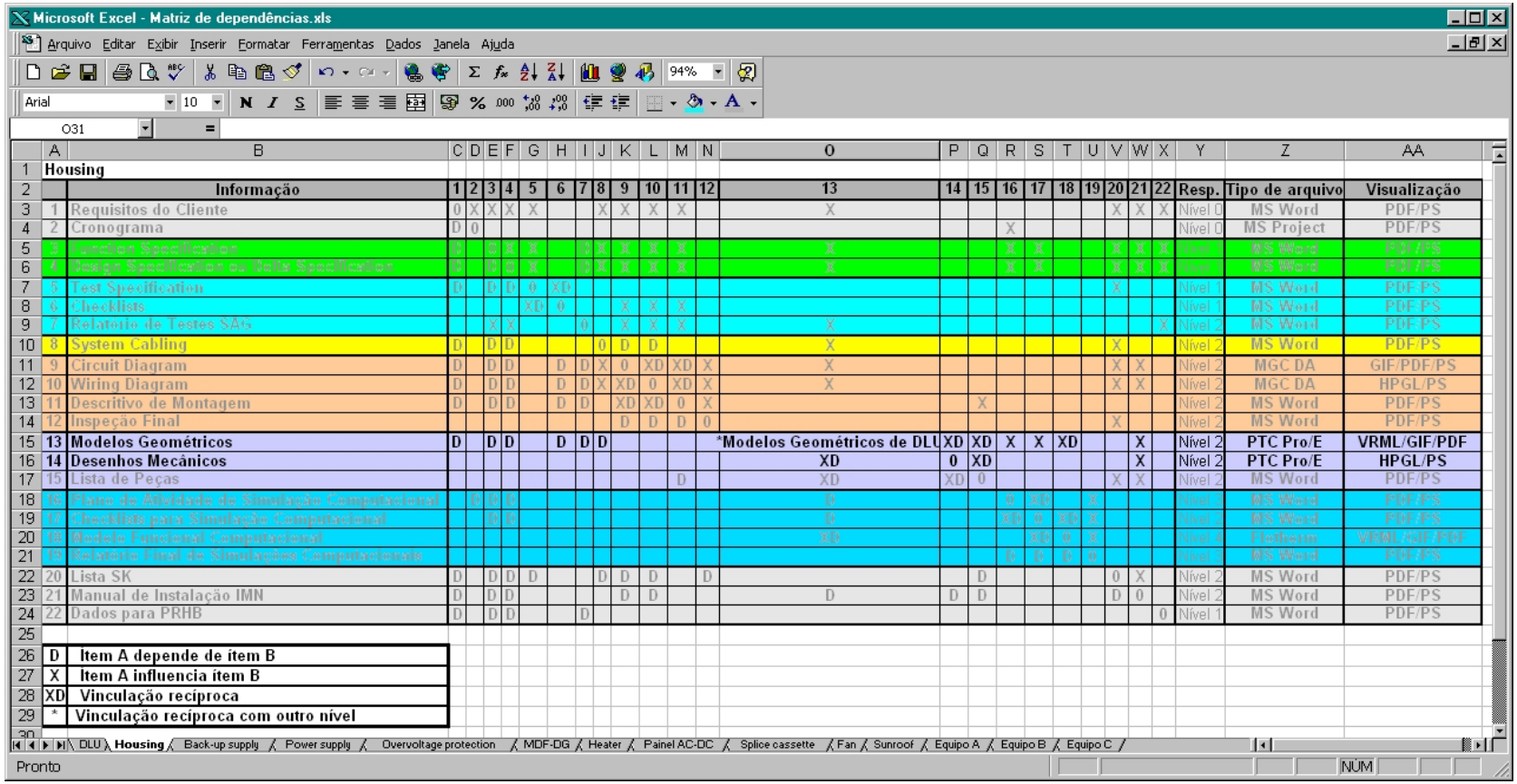
1	DLU																									
2		Informação	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	Resp.	Tipo de arquivo	Visualização
3	1	Requisitos do Cliente	0	X	X	X	X			X	X	X	X		X							X	X	X	Nível 0	MS Word	PDF/PS
4	2	Cronograma	D	0														X							Nível 0	MS Project	PDF/PS
5	3	Function Specification	D		0	X	X		D	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
6	4	Design Specification ou Delta Specification	D		D	0	X		D	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
7	5	Test Specification	D		D	D	0	XD														X			Nível 1	MS Word	PDF/PS
8	6	Checklists					XD	0			X	X	X												Nível 1	MS Word	PDF/PS
9	7	Relatório de Testes SAG			X	X			0		X	X	X		X									X	Nível 2	MS Word	PDF/PS
10	8	System Cabling	D		D	D				0	D	D			X							X			Nível 2	MS Word	PDF/PS
11	9	Circuit Diagram	D		D	D		D	D	X	*	XD	XD	X	X							X	X		Nível 2	MGC DA	GIF/PDF/PS
12	10	Wiring Diagram	D		D	D		D	D	X	XD	0	XD	X	X							X	X		Nível 2	MGC DA	HPGL/PS
13	11	Descritivo de Montagem	D		D	D		D	D		XD	XD	0	X			X								Nível 2	MS Word	PDF/PS
14	12	Inspeção Final									D	D	D	0								X			Nível 2	MS Word	PDF/PS
15	13	Modelos Geométricos	D		D	D		D	D	D	D	D			*	XD	XD	X	X	XD			X		Nível 2	PTC Pro/E	VRML/GIF/PDF
16	14	Desenhos Mecânicos													XD	0	XD						X		Nível 2	PTC Pro/E	HPGL/PS
17	15	Lista de Peças											D		XD	XD	0					X	X		Nível 2	MS Word	PDF/PS
18	16	Plano de Atividade de Simulação Computacional		D	D	D									D			0	XD		X				Nível 3	MS Word	PDF/PS
19	17	Checklists para Simulação Computacional			D	D									D			XD	0	XD	X				Nível 2	MS Word	PDF/PS
20	18	Modelo Funcional Computacional													XD				XD	0	X				Nível 4	Flotherm	VRML/GIF/PDF
21	19	Relatório Final de Simulações Computacionais																D	D	D	0				Nível 3	MS Word	PDF/PS
22	20	Lista SK	D		D	D	D			D	D	D		D			D					0	X		Nível 2	MS Word	PDF/PS
23	21	Manual de Instalação IMN	D		D	D					D	D			D	D	D					D	0		Nível 2	MS Word	PDF/PS
24	22	Dados para PRHB	D		D	D			D															0	Nível 1	MS Word	PDF/PS
25																											

D	Ítem A depende de ítem B
X	Ítem A influencia ítem B
XD	Vinculação recíproca
*	Vinculação recíproca com outro nível

DLU / Housing / Back-up supply / Power supply / Overvoltage protection / MDF-DG / Heater / Painel AC-DC / Splice cassette / Fan / Sunroof / Equipo A / Equipo B / Equipo C

Pronto

NUM



Microsoft Excel - Matriz de dependências.xls

Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Ajuda

Arial 10 — 94% — O31

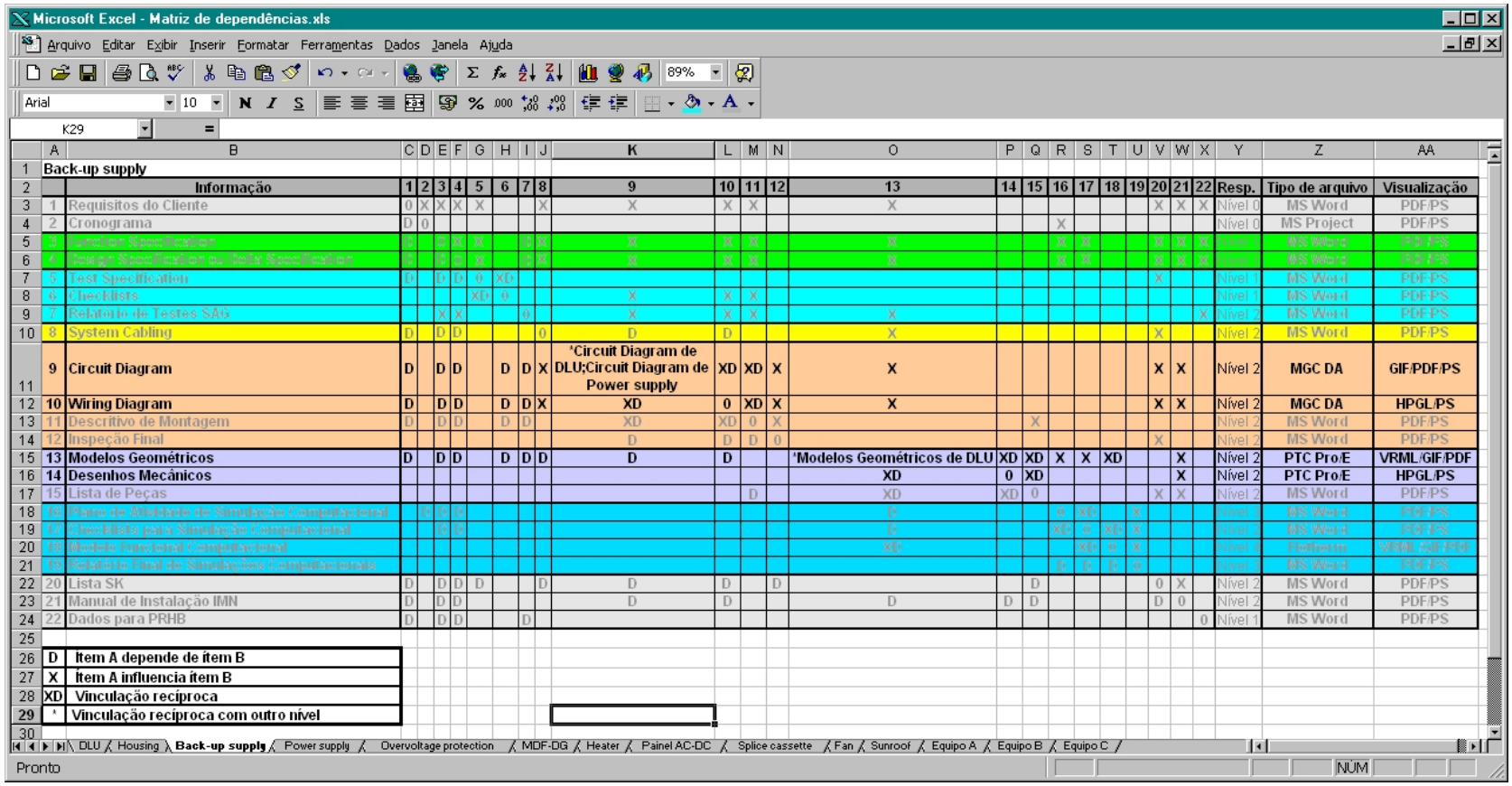
Housing

		Informação	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	Resp.	Tipo de arquivo	Visualização
3	1	Requisitos do Cliente	0	X	X	X	X			X	X	X	X		X							X	X	X	Nível 0	MS Word	PDF/PS
4	2	Cronograma	D	0														X							Nível 0	MS Project	PDF/PS
5	3	Function Specification	D		0	X	X		D	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
6	4	Design Specification ou Data Specification	D		D	0	X		D	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
7	5	Test Specification	D		D	D	0	XD							X							X			Nível 1	MS Word	PDF/PS
8	6	Checklists					XD	0			X	X	X												Nível 1	MS Word	PDF/PS
9	7	Relatório de Testes SADs			X	X			0		X	X	X		X									X	Nível 2	MS Word	PDF/PS
10	8	System Cabling	D		D	D				0	D	D			X							X			Nível 2	MS Word	PDF/PS
11	9	Circuit Diagram	D		D	D		D	D	X	0	XD	XD	X	X							X	X		Nível 2	MGC DA	GIF/PDF/PS
12	10	Wiring Diagram	D		D	D		D	D	X	XD	0	XD	X	X							X	X		Nível 2	MGC DA	HPGL/PS
13	11	Descritivo de Montagem	D		D	D		D	D		XD	XD	0	X			X								Nível 2	MS Word	PDF/PS
14	12	Inspeção Final									D	D	D	0								X			Nível 2	MS Word	PDF/PS
15	13	**Modelos Geométricos**	D		D	D		D	D	D					*Modelos Geométricos de DLU	XD	XD	X	X	XD			X		Nível 2	PTC Pro/E	VRML/GIF/PDF
16	14	**Desenhos Mecânicos**													XD	0	XD						X		Nível 2	PTC Pro/E	HPGL/PS
17	15	Lista de Peças											D		XD	XD	0					X	X		Nível 2	MS Word	PDF/PS
18	16	Plano de Atividade de Simulação Computacional		D	D	D									D			0	XD		X				Nível 3	MS Word	PDF/PS
19	17	Checklists para Simulação Computacional			D	D									D			XD	0	XD	X				Nível 3	MS Word	PDF/PS
20	18	Modelo Funcional Computacional													XD				XD	0	X				Nível 3	Fluiterm	VRML/GIF/PDF
21	19	Relatório Final de Simulações Computacionais																0	0	0	0				Nível 3	MS Word	PDF/PS
22	20	Lista SK	D		D	D	D			D	D	D		D			D					0	X		Nível 2	MS Word	PDF/PS
23	21	Manual de Instalação IMN	D		D	D				D	D				D	D	D					D	0		Nível 2	MS Word	PDF/PS
24	22	Dados para PRHB	D		D	D			D															0	Nível 1	MS Word	PDF/PS

D	Item A depende de ítem B
X	Item A influencia item B
XD	Vinculação recíproca
*	Vinculação recíproca com outro nível

DLU | Housing | Back-up supply | Power supply | Overvoltage protection | MDF-DG | Heater | Painel AC-DC | Splice cassette | Fan | Sunroof | Equipo A | Equipo B | Equipo C

Pronto — NÚM



Microsoft Excel - Matriz de dependências.xls

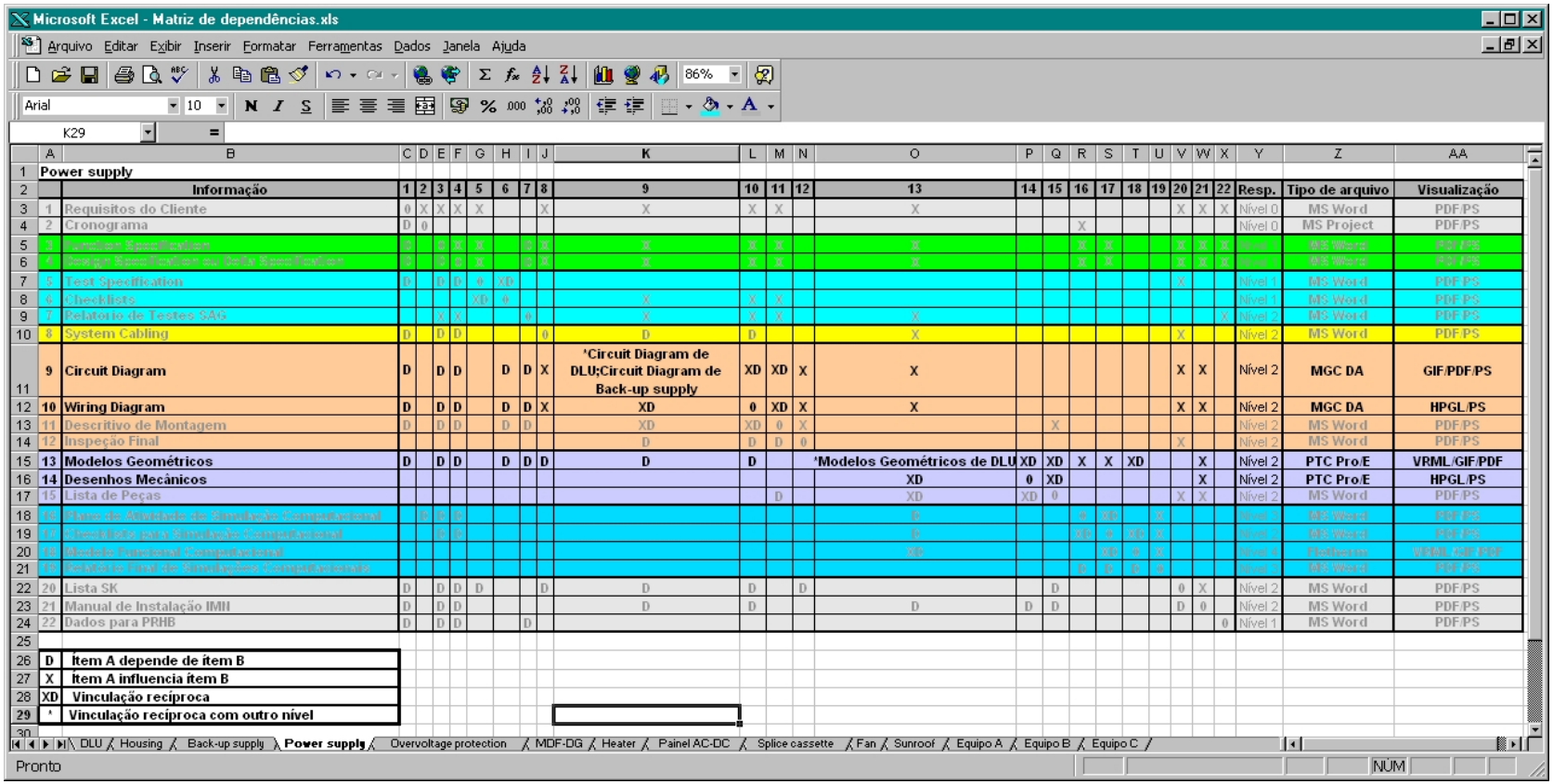
Back-up supply

		Informação	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	Resp.	Tipo de arquivo	Visualização
3	1	Requisitos do Cliente	0	X	X	X	X			X	X	X	X		X							X	X	X	Nível 0	MS Word	PDF/PS
4	2	Cronograma	D	0														X							Nível 0	MS Project	PDF/PS
5	3	Function Specification	D		0	X	X			X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
6	4	Design Specification ou Sub Specification	D		D	0	X			X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
7	5	Test Specification	D		D	D	0	XD														X			Nível 1	MS Word	PDF/PS
8	6	Checklists					XD	0			X	X	X												Nível 1	MS Word	PDF/PS
9	7	Relatorio de Testes SAG			X	X			0		X	X	X		X									X	Nível 2	MS Word	PDF/PS
10	8	System Cabling	D		D	D				0	D	D			X							X			Nível 2	MS Word	PDF/PS
11	9	Circuit Diagram	D		D	D		D	D	X	'Circuit Diagram de DLU;Circuit Diagram de Power supply	XD	XD	X	X							X	X		Nível 2	MGC DA	GIF/PDF/PS
12	10	Wiring Diagram	D		D	D		D	D	X	XD	0	XD	X	X							X	X		Nível 2	MGC DA	HPGL/PS
13	11	Descritivo de Montagem	D		D	D		D	D		XD	XD	0	X			X								Nível 2	MS Word	PDF/PS
14	12	Inspeção Final									D	D	D	0								X			Nível 2	MS Word	PDF/PS
15	13	Modelos Geométricos	D		D	D		D	D	D	D	D			'Modelos Geométricos de DLU	XD	XD	X	X	XD			X		Nível 2	PTC Pro/E	VRML/GIF/PDF
16	14	Desenhos Mecânicos													XD	0	XD						X		Nível 2	PTC Pro/E	HPGL/PS
17	15	Lista de Peças											D		XD	XD	0					X	X		Nível 2	MS Word	PDF/PS
18	16	Plano de Atividade de Simulação Computacional		D	D	D									D			D	XD		X				Nível 1	MS Word	PDF/PS
19	17	Checklists para Simulação Computacional			D	D									D			XD	0	XD	X				Nível 1	MS Word	PDF/PS
20	18	Modelo Funcional Computacional													XD				XD	0	X				Nível 1	Electronic	VRML/GIF/PDF
21	19	Relatorio Final de Simulações Computacionais													D			D	D	D	0				Nível 1	MS Word	PDF/PS
22	20	Lista SK	D		D	D	D			D	D	D		D			D					0	X		Nível 2	MS Word	PDF/PS
23	21	Manual de Instalação IMN	D		D	D					D	D			D	D	D					D	0		Nível 2	MS Word	PDF/PS
24	22	Dados para PRHB	D		D	D			D															0	Nível 1	MS Word	PDF/PS

D	Item A depende de item B
X	Item A influencia item B
XD	Vinculação recíproca
*	Vinculação recíproca com outro nível

DLU | Housing | Back-up supply | Power supply | Overvoltage protection | MDF-DG | Heater | Painel AC-DC | Splice cassette | Fan | Sunroof | Equipo A | Equipo B | Equipo C

Pronto



Microsoft Excel - Matriz de dependências.xls

Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Ajuda

Arial 10 — K29 =

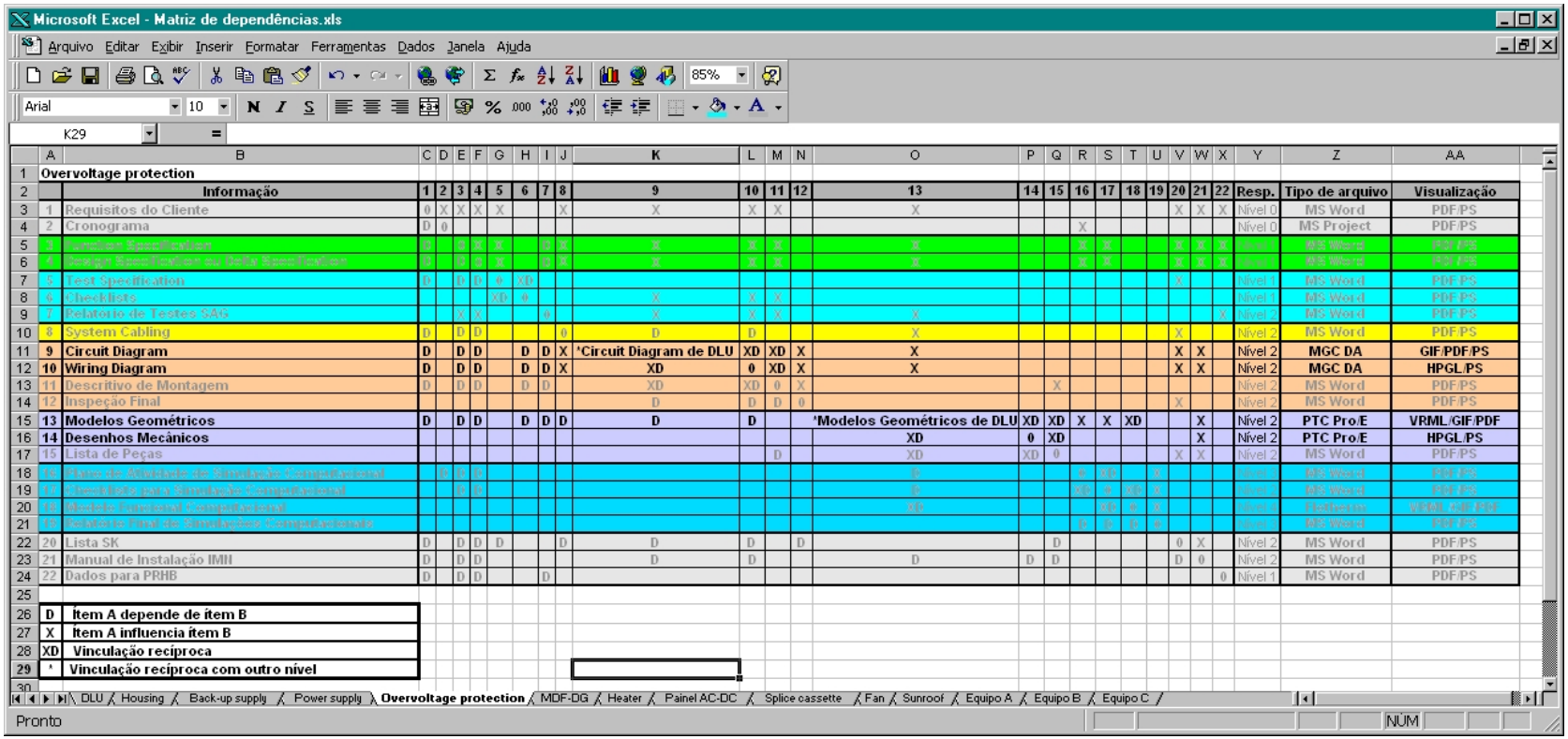
Power supply

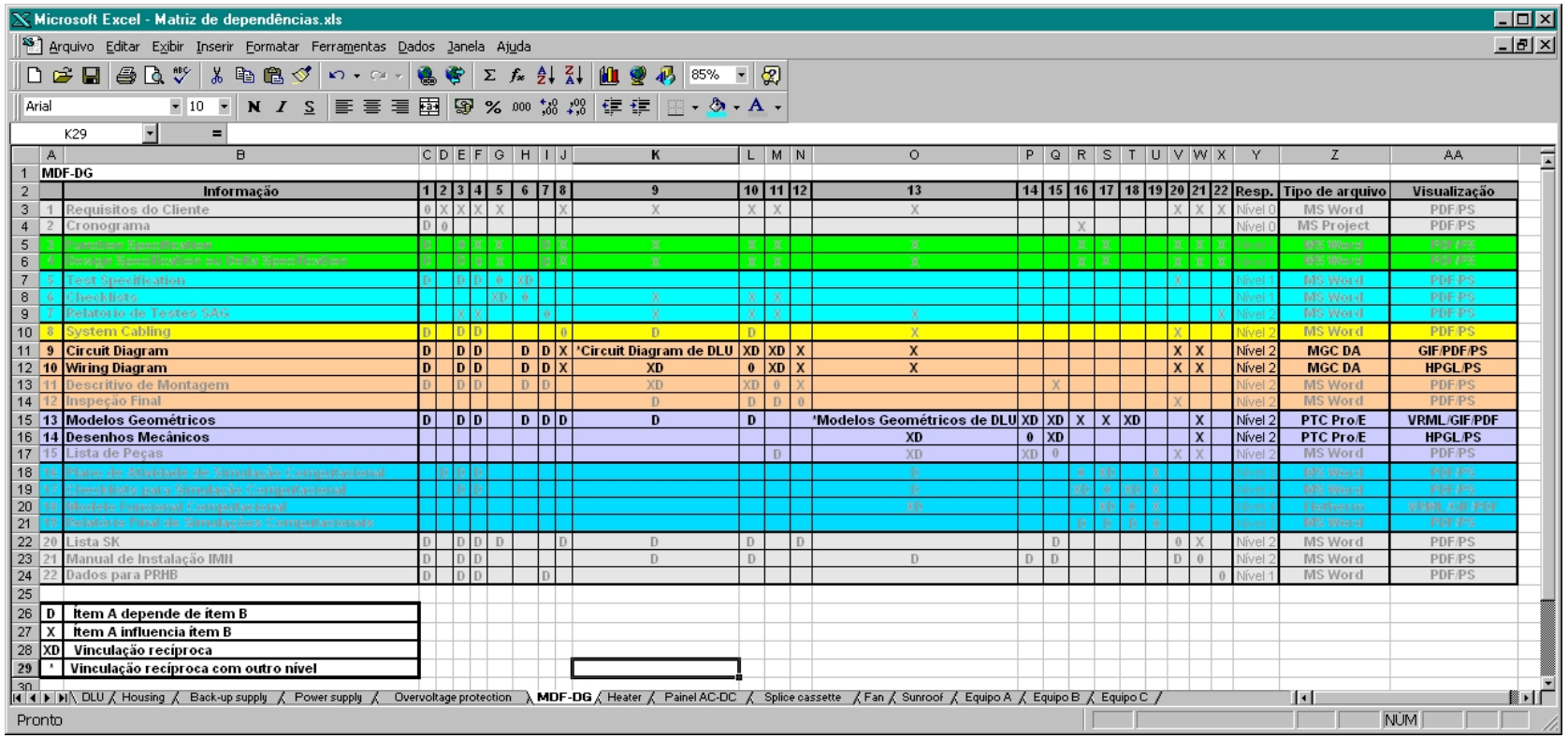
		Informação	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	Resp.	Tipo de arquivo	Visualização
3	1	Requisitos do Cliente	0	X	X	X	X			X	X	X	X		X							X	X	X	Nível 0	MS Word	PDF/PS
4	2	Cronograma	D	0														X							Nível 0	MS Project	PDF/PS
5	3	Function Specification	D		0	X	X			D	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
6	4	Design Specification ou Data Specification	D		D	X	X			D	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
7	5	Test Specification	D		D	D	0	XD														X			Nível 1	MS Word	PDF/PS
8	6	Checklists					XD	0			X	X	X												Nível 1	MS Word	PDF/PS
9	7	Relatorio de Testes SAG			X	X		0			X	X	X		X									X	Nível 2	MS Word	PDF/PS
10	8	System Cabling	D		D	D				0	D	D			X							X			Nível 2	MS Word	PDF/PS
11	9	Circuit Diagram	D		D	D		D	D	X	'Circuit Diagram de DLU;Circuit Diagram de Back-up supply	XD	XD	X	X							X	X		Nível 2	MGC DA	GIF/PDF/PS
12	10	Wiring Diagram	D		D	D		D	D	X	XD	0	XD	X	X							X	X		Nível 2	MGC DA	HPGL/PS
13	11	Descritivo de Montagem	D		D	D		D	D		XD	XD	0	X			X								Nível 2	MS Word	PDF/PS
14	12	Inspeção Final									D	D	D	0								X			Nível 2	MS Word	PDF/PS
15	13	Modelos Geométricos	D		D	D		D	D	D	D	D			'Modelos Geométricos de DLU	XD	XD	X	X	XD			X		Nível 2	PTC Pro/E	VRML/GIF/PDF
16	14	Desenhos Mecânicos													XD	0	XD						X		Nível 2	PTC Pro/E	HPGL/PS
17	15	Lista de Peças											D		XD	XD	0					X	X		Nível 2	MS Word	PDF/PS
18	16	Plano de Atividade de Simulação Computacional		D	D	D									D			X	XD	X					Nível 3	MS Word	PDF/PS
19	17	Checklists para Simulação Computacional			D	D									D			XD	X	XD	X				Nível 3	MS Word	PDF/PS
20	18	Modelo Funcional Computacional													XD			XD	X	X					Nível 4	Elsfluxus	VRML/GIF/PDF
21	19	Relatório final de Simulações Computacionais																X	X	X	X				Nível 3	MS Word	PDF/PS
22	20	Lista SK	D		D	D	D			D	D	D		D			D					0	X		Nível 2	MS Word	PDF/PS
23	21	Manual de Instalação IMII	D		D	D					D	D			D	D	D					D	0		Nível 2	MS Word	PDF/PS
24	22	Dados para PRHB	D		D	D			D															0	Nível 1	MS Word	PDF/PS

D	Ítem A depende de ítem B
X	Ítem A influencia ítem B
XD	Vinculação recíproca
'	Vinculação recíproca com outro nível

DLU | Housing | Back-up supply | Power supply | Overvoltage protection | MDF-DG | Heater | Painel AC-DC | Splice cassette | Fan | Sunroof | Equipo A | Equipo B | Equipo C

Pronto — NUM


Microsoft Excel - Matriz de dependências.xls
Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Ajuda
Arial 10
K29
Overvoltage protection
Informação
Resp. Tipo de arquivo Visualização
1 Requisitos do Cliente Nível 0 MS Word PDF/PS
2 Cronograma Nível 0 MS Project PDF/PS
3 Function Specification Nível 0 MS Word PDF/PS
4 Design Specification ou Delta Specification Nível 0 MS Word PDF/PS
5 Test Specification Nível 1 MS Word PDF/PS
6 Checklists Nível 1 MS Word PDF/PS
7 Relatorio de Testes SAS Nível 2 MS Word PDF/PS
8 System Cabling Nível 2 MS Word PDF/PS
9 Circuit Diagram Nível 2 MGC DA GIF/PDF/PS
10 Wiring Diagram Nível 2 MGC DA HPGL/PS
11 Descritivo de Montagem Nível 2 MS Word PDF/PS
12 Inspeção Final Nível 2 MS Word PDF/PS
13 Modelos Geométricos Nível 2 PTC Pro/E VRML/GIF/PDF
14 Desenhos Mecânicos Nível 2 PTC Pro/E HPGL/PS
15 Lista de Peças Nível 2 MS Word PDF/PS
16 Plano de Atividade de Simulação Computacional Nível 1 MS Word PDF/PS
17 Checklists para Simulação Computacional Nível 1 MS Word PDF/PS
18 Modelo Funcional Computacional Nível 1 Proprietário VRML/GIF/PDF
19 Relatório Final de Simulações Computacionais Nível 1 MS Word PDF/PS
20 Lista SK Nível 2 MS Word PDF/PS
21 Manual de Instalação IMH Nível 2 MS Word PDF/PS
22 Dados para PRHB Nível 1 MS Word PDF/PS
D Item A depende de item B
X Item A influencia item B
XD Vinculação reciproca
' Vinculação recíproca com outro nível
DLU Housing Back-up supply Power supply Overvoltage protection MDF-DG Heater Painel AC-DC Splice cassette Fan Sunroof Equipo A Equipo B Equipo C
Pronto
NUM



Microsoft Excel - Matriz de dependências.xls

Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Ajuda

Arial 10 — 85% — K29 =

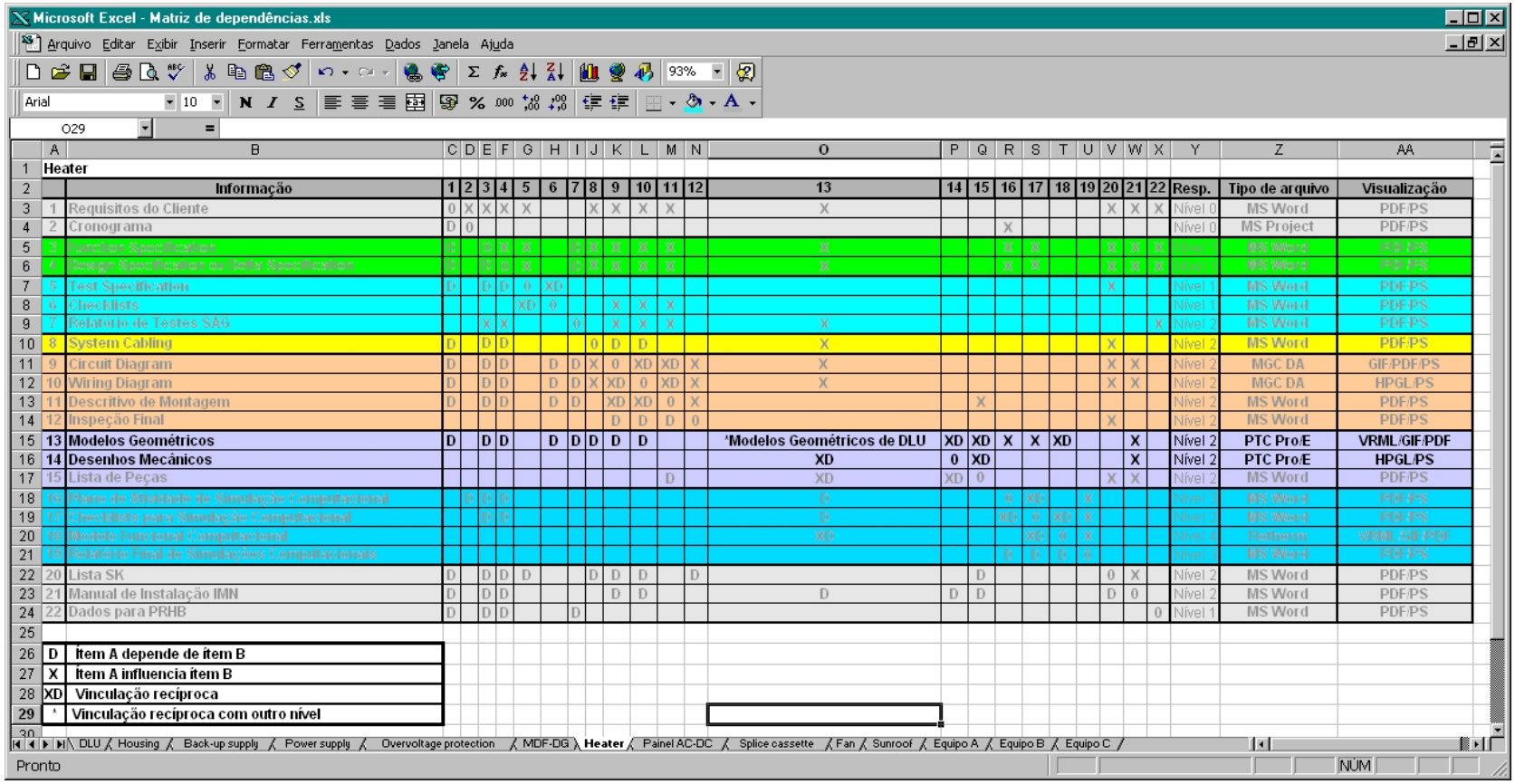
MDF-DG

		Informação	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	Resp.	Tipo de arquivo	Visualização
3	1	Requisitos do Cliente	0	X	X	X	X			X	X	X	X		X							X	X	X	Nível 0	MS Word	PDF/PS
4	2	Cronograma	D	0														X							Nível 0	MS Project	PDF/PS
5	3	Function Specification	D		0	X	X		D	X	X	X	X		X			X	X			X	D	X	Nível 1	MS Word	PDF/PS
6	4	Design Specification ou Data Specification	D		D	0	X		D	X	X	X	X		X			X	X			X	D	X	Nível 1	MS Word	PDF/PS
7	5	Test Specification	D		D	D	0	XD														X			Nível 1	MS Word	PDF/PS
8	6	Checklists					XD	0			X	X	X												Nível 1	MS Word	PDF/PS
9	7	Relatório de Testes SA6			X	X			0		X	X	X		X									X	Nível 2	MS Word	PDF/PS
10	8	System Cabling	D		D	D				0	D	D			X							X			Nível 2	MS Word	PDF/PS
11	9	Circuit Diagram	D		D	D		D	D	X	'Circuit Diagram de DLU	XD	XD	X	X							X	X		Nível 2	MGC DA	GIF/PDF/PS
12	10	Wiring Diagram	D		D	D		D	D	X	XD	0	XD	X	X							X	X		Nível 2	MGC DA	HPGL/PS
13	11	Descritivo de Montagem	D		D	D		D	D		XD	XD	0	X			X								Nível 2	MS Word	PDF/PS
14	12	Inspeção Final									D	D	D	0								X			Nível 2	MS Word	PDF/PS
15	13	Modelos Geométricos	D		D	D		D	D	D	D	D			'Modelos Geométricos de DLU	XD	XD	X	X	XD			X			Nível 2	PTC Pro/E	VRML/GIF/PDF
16	14	Desenhos Mecânicos													XD	0	XD						X		Nível 2	PTC Pro/E	HPGL/PS
17	15	Lista de Peças											D		XD	XD	0					X	X		Nível 2	MS Word	PDF/PS
18	16	Plano de Atividade de Simulação Computacional		D	D	D						D						#	XD		X				Nível 1	MS Word	PDF/PS
19	17	Checklists para Simulação Computacional			D	D						D						XD	#	XD	X				Nível 1	MS Word	PDF/PS
20	18	Modelo Funcional Computacional										XD							XD	#	X				Nível 1	Flowmaster	VRML/GIF/PDF
21	19	Relatório final de Simulações Computacionais																	D	D	#				Nível 1	MS Word	PDF/PS
22	20	Lista SK	D		D	D	D			D	D	D		D			D					0	X		Nível 2	MS Word	PDF/PS
23	21	Manual de Instalação IMN	D		D	D					D	D			D	D	D					D	0		Nível 2	MS Word	PDF/PS
24	22	Dados para PRHB	D		D	D			D															0	Nível 1	MS Word	PDF/PS

D	Ítem A depende de ítem B
X	Ítem A influencia ítem B
XD	Vinculação reciproca
'	Vinculação reciproca com outro nível

DLU | Housing | Back-up supply | Power supply | Overvoltage protection | MDF-DG | Heater | Painel AC-DC | Splice cassette | Fan | Sunroof | Equipo A | Equipo B | Equipo C

Pronto — NUM



Microsoft Excel - Matriz de dependências.xls

Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Ajuda

Heater

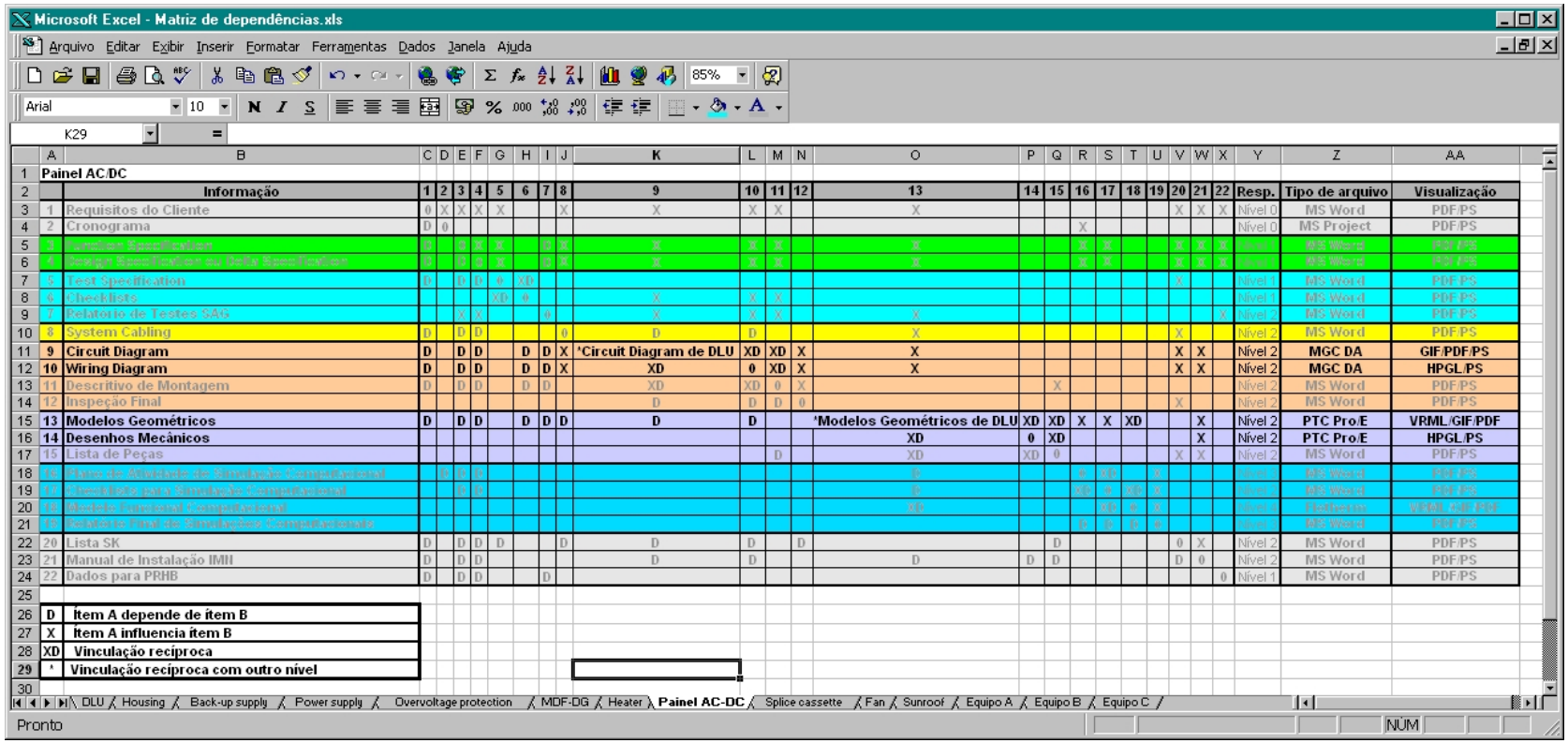
		Informação	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	Resp.	Tipo de arquivo	Visualização
3	1	Requisitos do Cliente	0	X	X	X	X			X	X	X	X		X							X	X	X	Nível 0	MS Word	PDF/PS
4	2	Cronograma	D	0														X							Nível 0	MS Project	PDF/PS
5	3	Function Specification	D		D	X	X		D	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
6	4	Design Specification ou Data Specification	D		D	D	X		D	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
7	5	Test Specification	D		D	D	0	XD														X			Nível 1	MS Word	PDF/PS
8	6	Checklists					XD	0			X	X	X												Nível 1	MS Word	PDF/PS
9	7	Relatório de Testes SAG			X	X			0		X	X	X		X									X	Nível 2	MS Word	PDF/PS
10	8	System Cabling	D		D	D				0	D	D			X							X			Nível 2	MS Word	PDF/PS
11	9	Circuit Diagram	D		D	D		D	D	X	0	XD	XD	X	X							X	X		Nível 2	MGC DA	GIF/PDF/PS
12	10	Wiring Diagram	D		D	D		D	D	X	XD	0	XD	X	X							X	X		Nível 2	MGC DA	HPGL/PS
13	11	Descritivo de Montagem	D		D	D		D	D		XD	XD	0	X			X								Nível 2	MS Word	PDF/PS
14	12	Inspeção Final									D	D	D	0								X			Nível 2	MS Word	PDF/PS
15	13	**Modelos Geométricos**	D		D	D		D	D	D	D	D			'Modelos Geométricos de DLU	XD	XD	X	X	XD			X		Nível 2	PTC Pro/E	VRML/GIF/PDF
16	14	**Desenhos Mecânicos**													XD	0	XD						X		Nível 2	PTC Pro/E	HPGL/PS
17	15	Lista de Peças											D		XD	XD	0					X	X		Nível 2	MS Word	PDF/PS
18	16	Plano de Atividade de Simulação Computacional		D	D	D									C			0	XD		X				Nível 3	MS Word	PDF/PS
19	17	Checklists para Simulação Computacional			D	D									C			XD	0	XD	X				Nível 3	MS Word	PDF/PS
20	18	Modelo Estrutural Computacional													XC				XD	0	X				Nível 4	Ansys	VRML/GIF/PDF
21	19	Relatório Final de Simulações Computacionais																D	D	D	0				Nível 3	MS Word	PDF/PS
22	20	Lista SK	D		D	D	D			D	D	D		D			D					0	X		Nível 2	MS Word	PDF/PS
23	21	Manual de Instalação IMN	D		D	D					D	D			D	D	D					D	0		Nível 2	MS Word	PDF/PS
24	22	Dados para PRHB	D		D	D			D															0	Nível 1	MS Word	PDF/PS

D	Item A depende de item B
X	Item A influencia item B
XD	Vinculação recíproca
*	Vinculação recíproca com outro nível

DLU | Housing | Back-up supply | Power supply | Overvoltage protection | MDF-DG | Heater | Painel AC-DC | Splice cassette | Fan | Sunroof | Equipo A | Equipo B | Equipo C

Pronto

NUM



Microsoft Excel - Matriz de dependências.xls

Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Ajuda

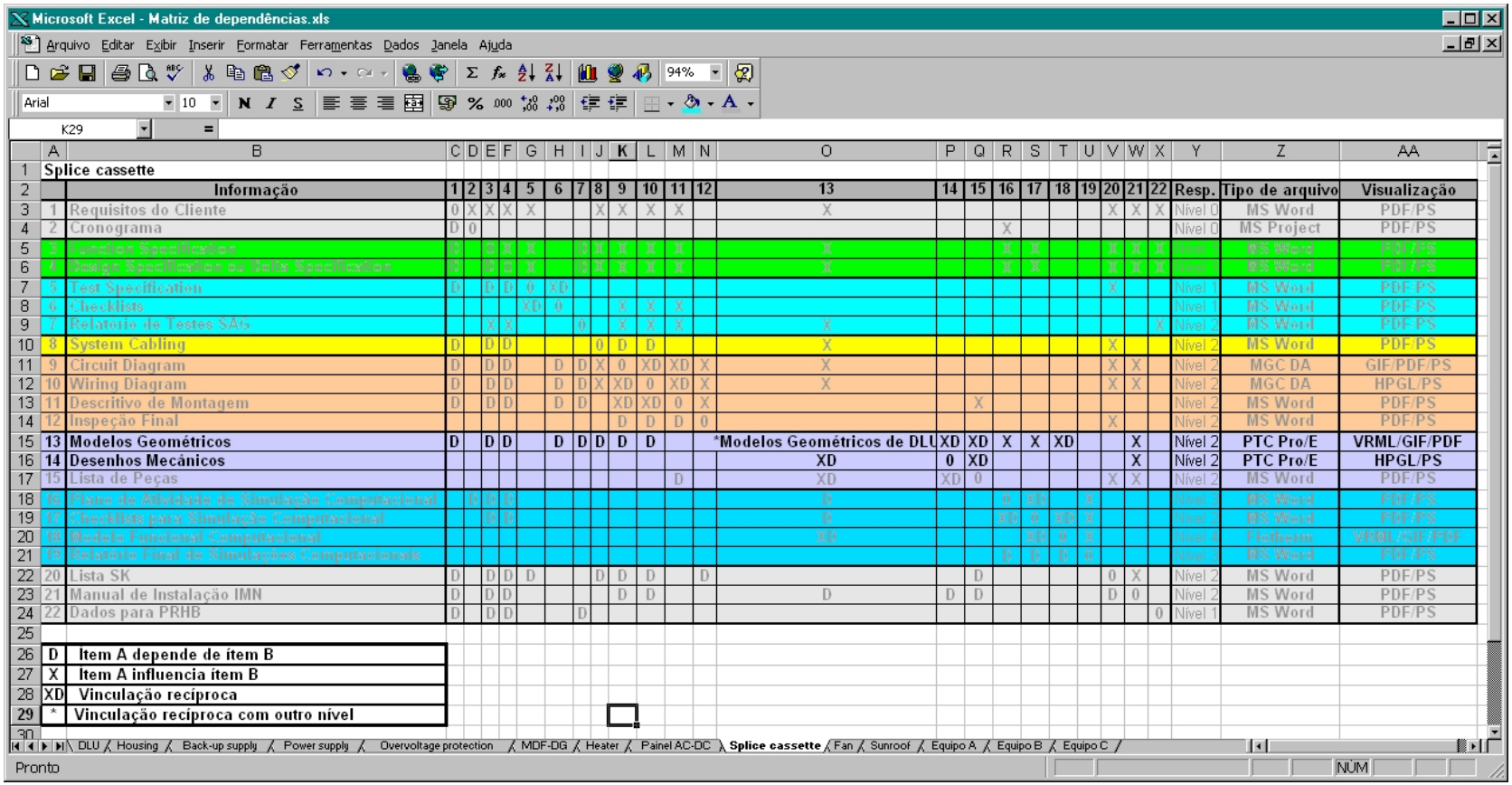
Arial 10 — 85%

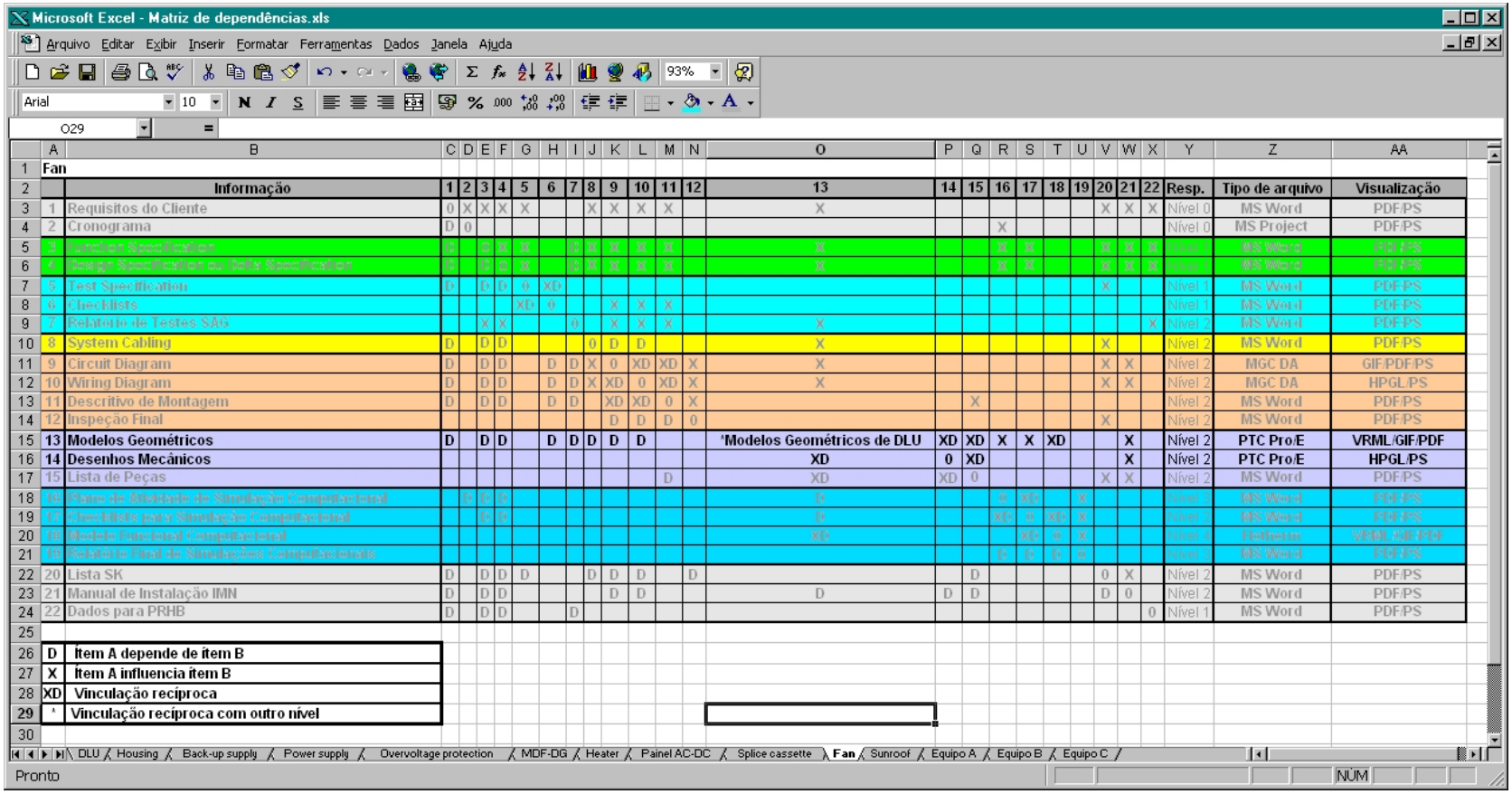
K29 =

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1	Painel AC/DC																										
2		Informação	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	Resp.	Tipo de arquivo	Visualização
3	1	Requisitos do Cliente	0	X	X	X	X			X	X	X	X		X							X	X	X	Nível 0	MS Word	PDF/PS
4	2	Cronograma	D	0														X							Nível 0	MS Project	PDF/PS
5	3	Function Specification	D		0	X	X		D	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
6	4	Design Specification or Data Specification	D		D	0	X		0	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
7	5	Test Specification	D		D	D	0	XD														X			Nível 1	MS Word	PDF/PS
8	6	Checklists					XD	0			X	X	X												Nível 1	MS Word	PDF/PS
9	7	Relatorio de Testes SAS			X	X			0		X	X	X		X									X	Nível 2	MS Word	PDF/PS
10	8	System Cabling	D		D	D				0	D	D			X							X			Nível 2	MS Word	PDF/PS
11	9	Circuit Diagram	D		D	D		D	D	X	'Circuit Diagram de DLU	XD	XD	X	X							X	X		Nível 2	MGC DA	GIF/PDF/PS
12	10	Wiring Diagram	D		D	D		D	D	X	XD	0	XD	X	X							X	X		Nível 2	MGC DA	HPGL/PS
13	11	Descritivo de Montagem	D		D	D		D	D		XD	XD	0	X			X								Nível 2	MS Word	PDF/PS
14	12	Inspeção Final									D	D	D	0								X			Nível 2	MS Word	PDF/PS
15	13	Modelos Geométricos	D		D	D		D	D	D	D	D			'Modelos Geométricos de DLU	XD	XD	X	X	XD			X		Nível 2	PTC Pro/E	VRML/GIF/PDF
16	14	Desenhos Mecânicos													XD	0	XD						X		Nível 2	PTC Pro/E	HPGL/PS
17	15	Lista de Peças											D		XD	XD	0					X	X		Nível 2	MS Word	PDF/PS
18	16	Plano de Atividade de Simulação Computacional		D	D	D									D			D	XD		X				Nível 1	MS Word	PDF/PS
19	17	Checklists para Simulação Computacional			D	D									D			XD	0	XD	X				Nível 2	MS Word	PDF/PS
20	18	Modelo Funcional Computacional													XD				XD	0	X				Nível 2	Fluxview	VRML/GIF/PDF
21	19	Relatorio Final de Simulação Computacional																X	X	D	0				Nível 1	MS Word	PDF/PS
22	20	Lista SK	D		D	D	D			D	D	D		D			D					0	X		Nível 2	MS Word	PDF/PS
23	21	Manual de Instalação IMN	D		D	D					D	D			D	D	D					D	0		Nível 2	MS Word	PDF/PS
24	22	Dados para PRHB	D		D	D			D															0	Nível 1	MS Word	PDF/PS
25																											
26	D	Item A depende de item B																									
27	X	Item A influencia item B																									
28	XD	Vinculação recíproca																									
29	'	Vinculação recíproca com outro nível																									
30																											

DLU / Housing / Back-up supply / Power supply / Overvoltage protection / MDF-DG / Heater / Painel AC-DC / Splice cassette / Fan / Sunroof / Equipo A / Equipo B / Equipo C /

Pronto NUM


Microsoft Excel - Matriz de dependências.xls
Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Ajuda
Arial 10
K29
Splice cassette
Informação
Modelos Geométricos
Desenhos Mecânicos
Lista de Peças
Lista SK
Manual de Instalação IMN
Dados para PRHB
D Item A depende de item B
X Item A influencia item B
XD Vinculação recíproca
* Vinculação recíproca com outro nível
DLU Housing Back-up supply Power supply Overvoltage protection MDF-DG Heater Painel AC-DC Splice cassette Fan Sunroof Equipo A Equipo B Equipo C
Pronto
NÚM



Microsoft Excel - Matriz de dependências.xls

Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Ajuda

Fan

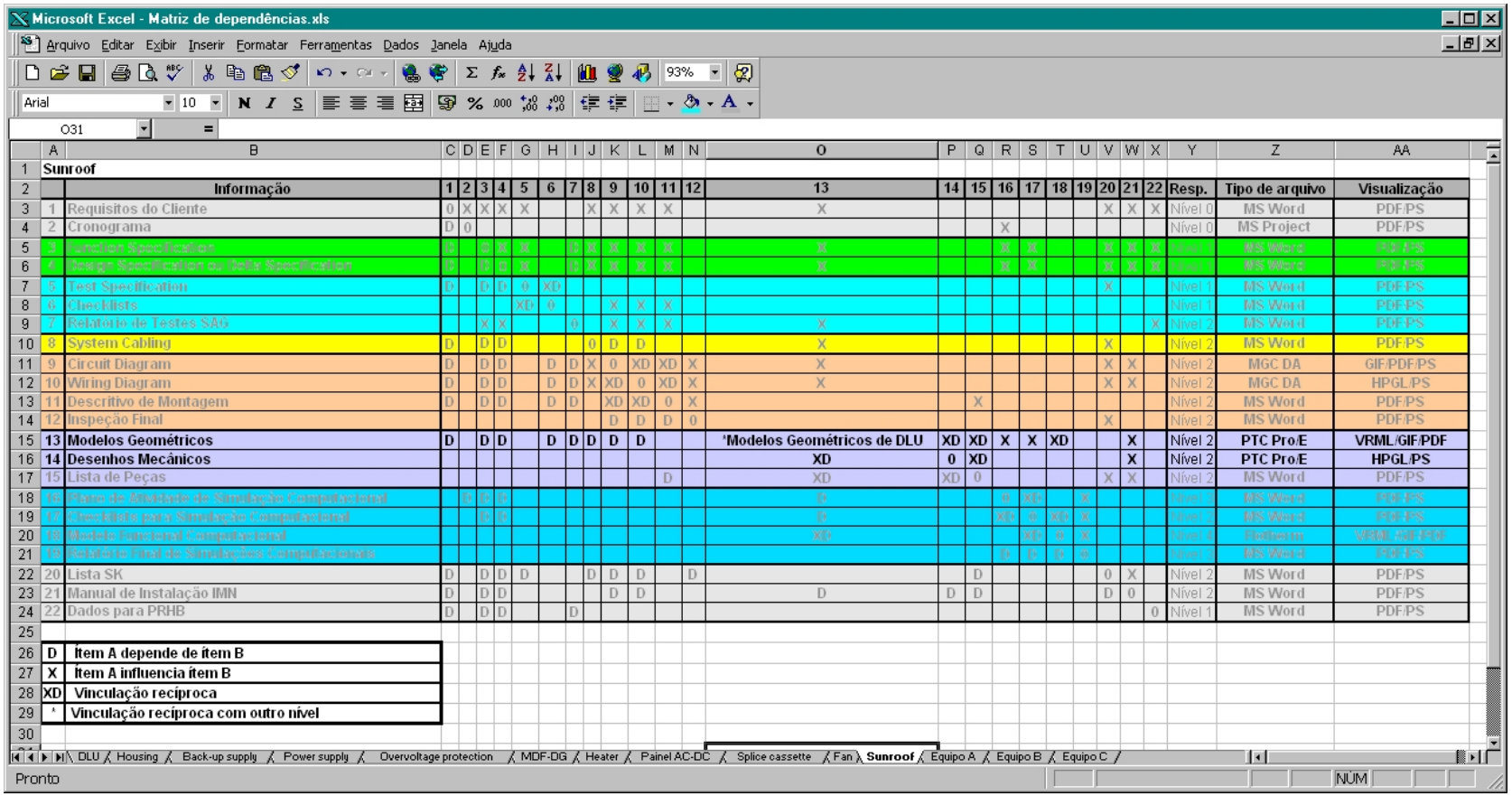
		Informação	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	Resp.	Tipo de arquivo	Visualização
3	1	Requisitos do Cliente	0	X	X	X	X			X	X	X	X		X							X	X	X	Nível 0	MS Word	PDF/PS
4	2	Cronograma	D	0														X							Nível 0	MS Project	PDF/PS
5	3	Function Specification	D		0	X	X		0	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
6	4	Design Specification ou Solid Specification	D		D	0	X		0	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
7	5	Test Specification	D		D	D	0	XD														X			Nível 1	MS Word	PDF/PS
8	6	Checklists					XD	0			X	X	X												Nível 1	MS Word	PDF/PS
9	7	Relatório de Testes SAG			X	X			0		X	X	X		X									X	Nível 2	MS Word	PDF/PS
10	8	System Cabling	D		D	D				0	D	D			X							X			Nível 2	MS Word	PDF/PS
11	9	Circuit Diagram	D		D	D		D	D	X	0	XD	XD	X	X							X	X		Nível 2	MGC DA	GIF/PDF/PS
12	10	Wiring Diagram	D		D	D		D	D	X	XD	0	XD	X	X							X	X		Nível 2	MGC DA	HPGL/PS
13	11	Descritivo de Montagem	D		D	D		D	D		XD	XD	0	X			X								Nível 2	MS Word	PDF/PS
14	12	Inspeção Final									D	D	D	0								X			Nível 2	MS Word	PDF/PS
15	13	**Modelos Geométricos**	D		D	D		D	D	D	D	D			'Modelos Geométricos de DLU	XD	XD	X	X	XD			X		Nível 2	PTC Pro/E	VRML/GIF/PDF
16	14	**Desenhos Mecânicos**													XD	0	XD						X		Nível 2	PTC Pro/E	HPGL/PS
17	15	Lista de Peças											D		XD	XD	0					X	X		Nível 2	MS Word	PDF/PS
18	16	Plano de Atividade de Simulação Computacional		D	D	D									D			X	XD		X				Nível 1	MS Word	PDF/PS
19	17	Checklists para Simulação Computacional			D	D									D			XD	X	XD	X				Nível 1	MS Word	PDF/PS
20	18	Modelo Funcional Computacional													XD				XD	X	X				Nível 1	Matlab/etc	VRML/GIF/PDF
21	19	Relatório Final de Simulações Computacionais																D	X	X	X				Nível 1	MS Word	PDF/PS
22	20	Lista SK	D		D	D	D			D	D	D		D			D					0	X		Nível 2	MS Word	PDF/PS
23	21	Manual de Instalação IMN	D		D	D				D	D				D	D	D					D	0		Nível 2	MS Word	PDF/PS
24	22	Dados para PRHB	D		D	D			D															0	Nível 1	MS Word	PDF/PS

D	Item A depende de item B
X	Item A influencia item B
XD	Vinculação recíproca
'	Vinculação recíproca com outro nível

DLU | Housing | Back-up supply | Power supply | Overvoltage protection | MDF-DG | Heater | Painel AC-DC | Splice cassette | Fan | Sunroof | Equipo A | Equipo B | Equipo C

Pronto

NÚM



Microsoft Excel - Matriz de dependências.xls

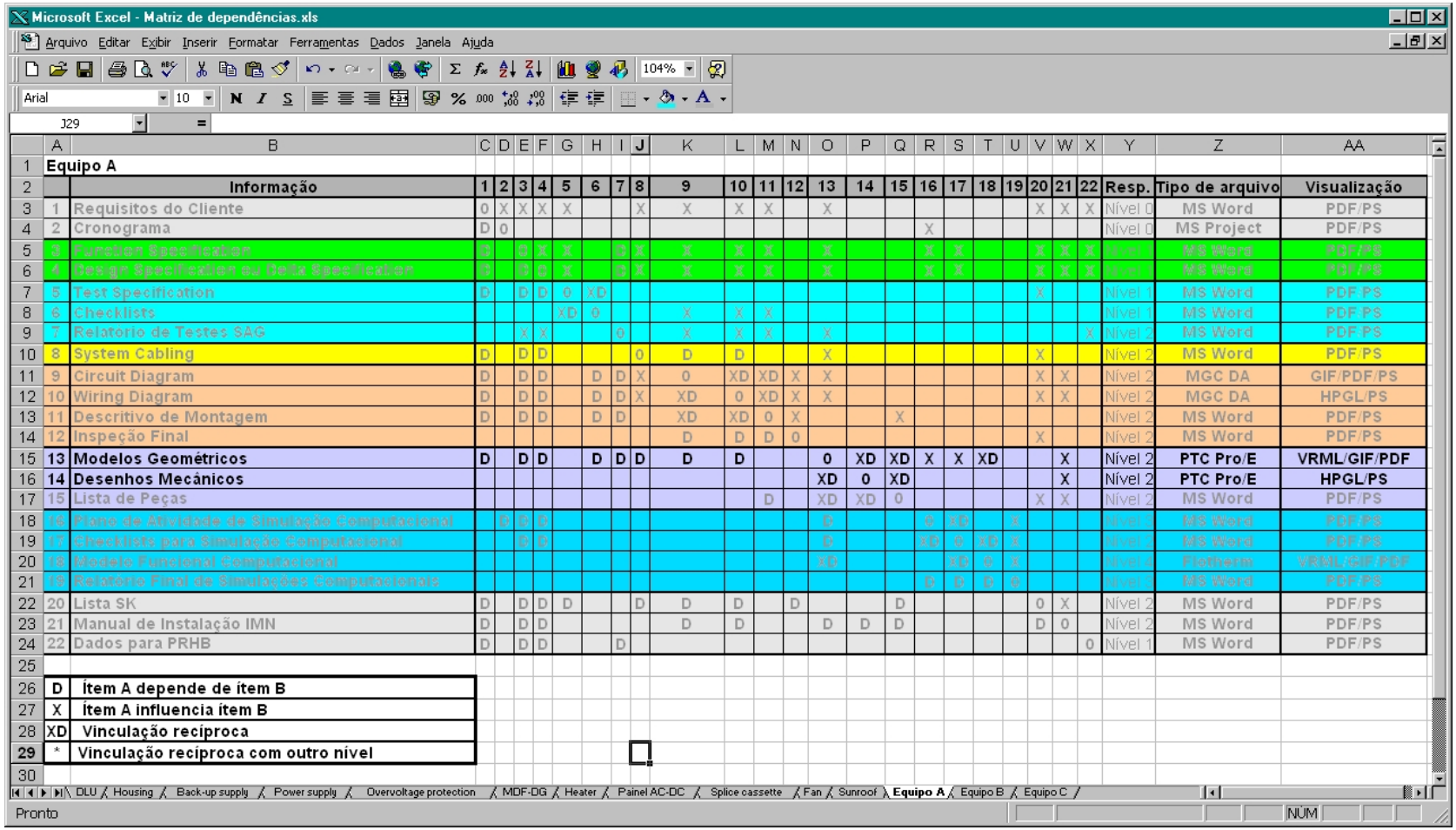
Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Ajuda

Arial 10 — 93% — O31 =

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1	Sunroof																										
2		Informação	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	Resp.	Tipo de arquivo	Visualização
3	1	Requisitos do Cliente	0	X	X	X	X			X	X	X	X		X							X	X	X	Nível 0	MS Word	PDF/PS
4	2	Cronograma	D	0														X							Nível 0	MS Project	PDF/PS
5	3	Function Specification	D		0	X	X		D	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
6	4	Design Specification or Data Specification	D		D	0	X		D	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
7	5	Test Specification	D		D	D	0	XD														X			Nível 1	MS Word	PDF/PS
8	6	Checklists					XD	0			X	X	X												Nível 1	MS Word	PDF/PS
9	7	Relatório de Testes SAG			X	X			0		X	X	X		X									X	Nível 2	MS Word	PDF/PS
10	8	System Cabling	D		D	D				0	D	D			X							X			Nível 2	MS Word	PDF/PS
11	9	Circuit Diagram	D		D	D		D	D	X	0	XD	XD	X	X							X	X		Nível 2	MGC DA	GIF/PDF/PS
12	10	Wiring Diagram	D		D	D		D	D	X	XD	0	XD	X	X							X	X		Nível 2	MGC DA	HPGL/PS
13	11	Descritivo de Montagem	D		D	D		D	D		XD	XD	0	X			X								Nível 2	MS Word	PDF/PS
14	12	Inspeção Final									D	D	D	0								X			Nível 2	MS Word	PDF/PS
15	13	**Modelos Geométricos**	D		D	D		D	D	D	D	D			*Modelos Geométricos de DLU	XD	XD	X	X	XD			X		Nível 2	PTC Pro/E	VRML/GIF/PDF
16	14	**Desenhos Mecânicos**													XD	0	XD						X		Nível 2	PTC Pro/E	HPGL/PS
17	15	Lista de Peças											D		XD	XD	0					X	X		Nível 2	MS Word	PDF/PS
18	16	Plano de Atividade de Simulação Computacional		D	D	D									D			0	XD		X				Nível 2	MS Word	PDF/PS
19	17	Checklists para Simulação Computacional			D	D									D			XD	0	XD	X				Nível 2	MS Word	PDF/PS
20	18	Modelo Funcional Computacional													XD				XD	0	X				Nível 2	Fluidsim	VRML/GIF/PDF
21	19	Relatório Final de Simulações Computacionais													D			D	D	D	0				Nível 2	MS Word	PDF/PS
22	20	Lista SK	D		D	D	D			D	D	D		D			D					0	X		Nível 2	MS Word	PDF/PS
23	21	Manual de Instalação IMN	D		D	D					D	D			D	D	D					D	0		Nível 2	MS Word	PDF/PS
24	22	Dados para PRHB	D		D	D			D															0	Nível 1	MS Word	PDF/PS
25																											
26	D	Item A depende de item B																									
27	X	Item A influencia item B																									
28	XD	Vinculação recíproca																									
29	*	Vinculação recíproca com outro nível																									
30																											

DLU | Housing | Back-up supply | Power supply | Overvoltage protection | MDF-DG | Heater | Painel AC-DC | Splice cassette | Fan | Sunroof | Equipo A | Equipo B | Equipo C

Pronto — NUM



Microsoft Excel - Matriz de dependências.xls

Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Ajuda

Arial 10 104%

J29 =

Equipo A

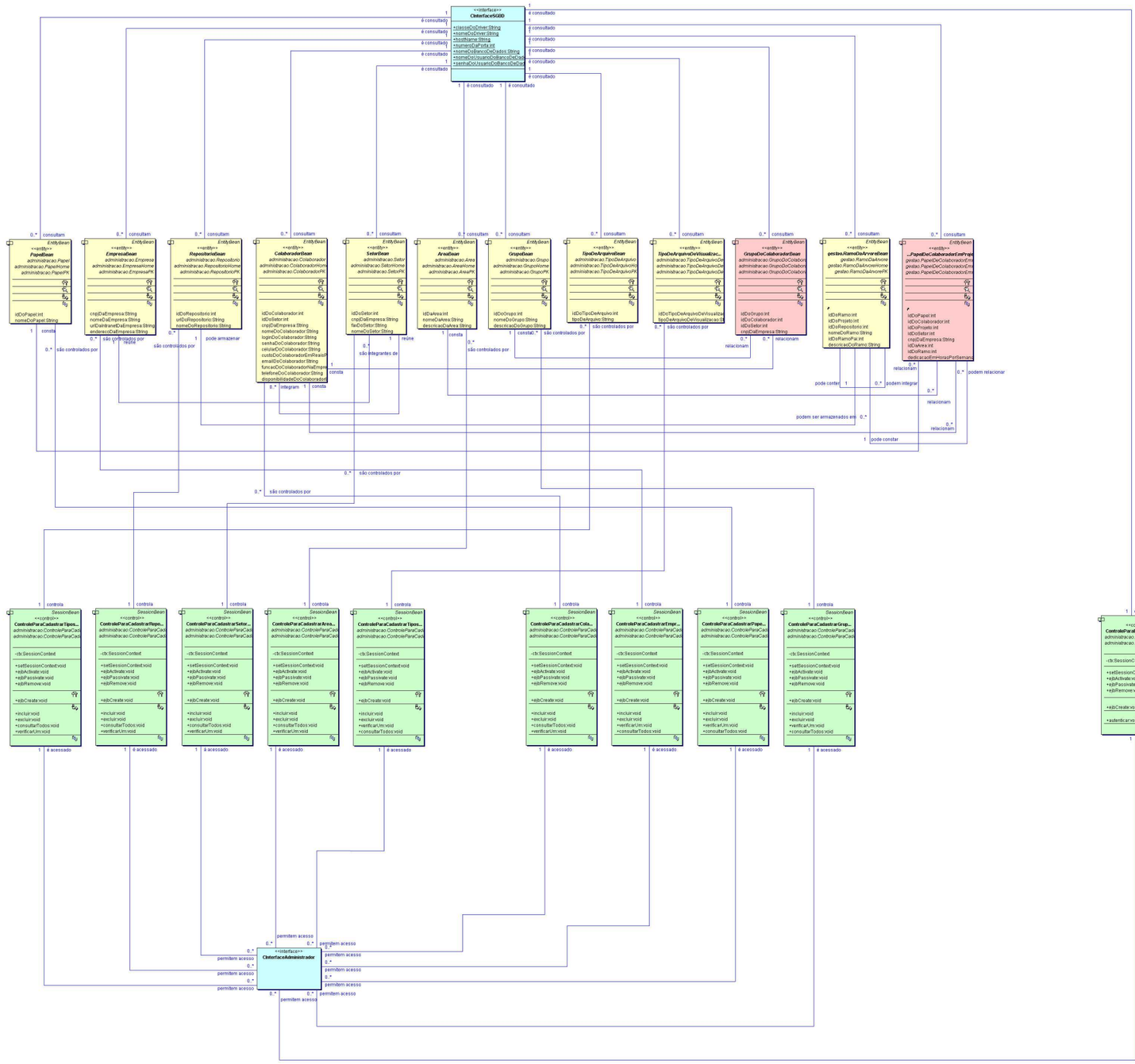
		Informação	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	Resp.	Tipo de arquivo	Visualização
3	1	Requisitos do Cliente	0	X	X	X	X			X	X	X	X		X							X	X	X	Nível 0	MS Word	PDF/PS
4	2	Cronograma	D	0														X							Nível 0	MS Project	PDF/PS
5	3	Function Specification	D		D	X	X		D	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
6	4	Design Specification ou Code Specification	D		D	D	X		D	X	X	X	X		X			X	X			X	X	X	Nível 1	MS Word	PDF/PS
7	5	Test Specification	D		D	D	0	XD														X			Nível 1	MS Word	PDF/PS
8	6	Checklists					XD	0			X	X	X												Nível 1	MS Word	PDF/PS
9	7	Relatório de Testes SAG			X	X			0		X	X	X		X									X	Nível 2	MS Word	PDF/PS
10	8	System Cabling	D		D	D				0	D	D			X							X			Nível 2	MS Word	PDF/PS
11	9	Circuit Diagram	D		D	D		D	D	X	0	XD	XD	X	X							X	X		Nível 2	MGC DA	GIF/PDF/PS
12	10	Wiring Diagram	D		D	D		D	D	X	XD	0	XD	X	X							X	X		Nível 2	MGC DA	HPGL/PS
13	11	Descritivo de Montagem	D		D	D		D	D		XD	XD	0	X			X								Nível 2	MS Word	PDF/PS
14	12	Inspeção Final									D	D	D	0								X			Nível 2	MS Word	PDF/PS
15	13	**Modelos Geométricos**	**D**		**D**	**D**		**D**	**D**	**D**	**D**	**D**			**0**	**XD**	**XD**	**X**	**X**	**XD**			**X**		Nível 2	**PTC Pro/E**	**VRML/GIF/PDF**
16	14	**Desenhos Mecânicos**													**XD**	**0**	**XD**						**X**		Nível 2	**PTC Pro/E**	**HPGL/PS**
17	15	Lista de Peças											D		XD	XD	0					X	X		Nível 2	MS Word	PDF/PS
18	16	Plano de Atividade de Simulação Computacional		D	D	D									D			D	XD		X				Nível 2	MS Word	PDF/PS
19	17	Checklists para Simulação Computacional			D	D									D			XD	0	XD	X				Nível 2	MS Word	PDF/PS
20	18	Modelo Funcional Computacional													XD				XD	0	X				Nível 2	Flotherm	VRML/GIF/PDF
21	19	Relatório Final de Simulações Computacionais																D	0	0	0				Nível 2	MS Word	PDF/PS
22	20	Lista SK	D		D	D	D			D	D	D		D			D					0	X		Nível 2	MS Word	PDF/PS
23	21	Manual de Instalação IMN	D		D	D					D	D			D	D	D					D	0		Nível 2	MS Word	PDF/PS
24	22	Dados para PRHB	D		D	D			D															0	Nível 1	MS Word	PDF/PS

D	Ítem A depende de ítem B
X	Ítem A influencia ítem B
XD	Vinculação recíproca
*	Vinculação recíproca com outro nível

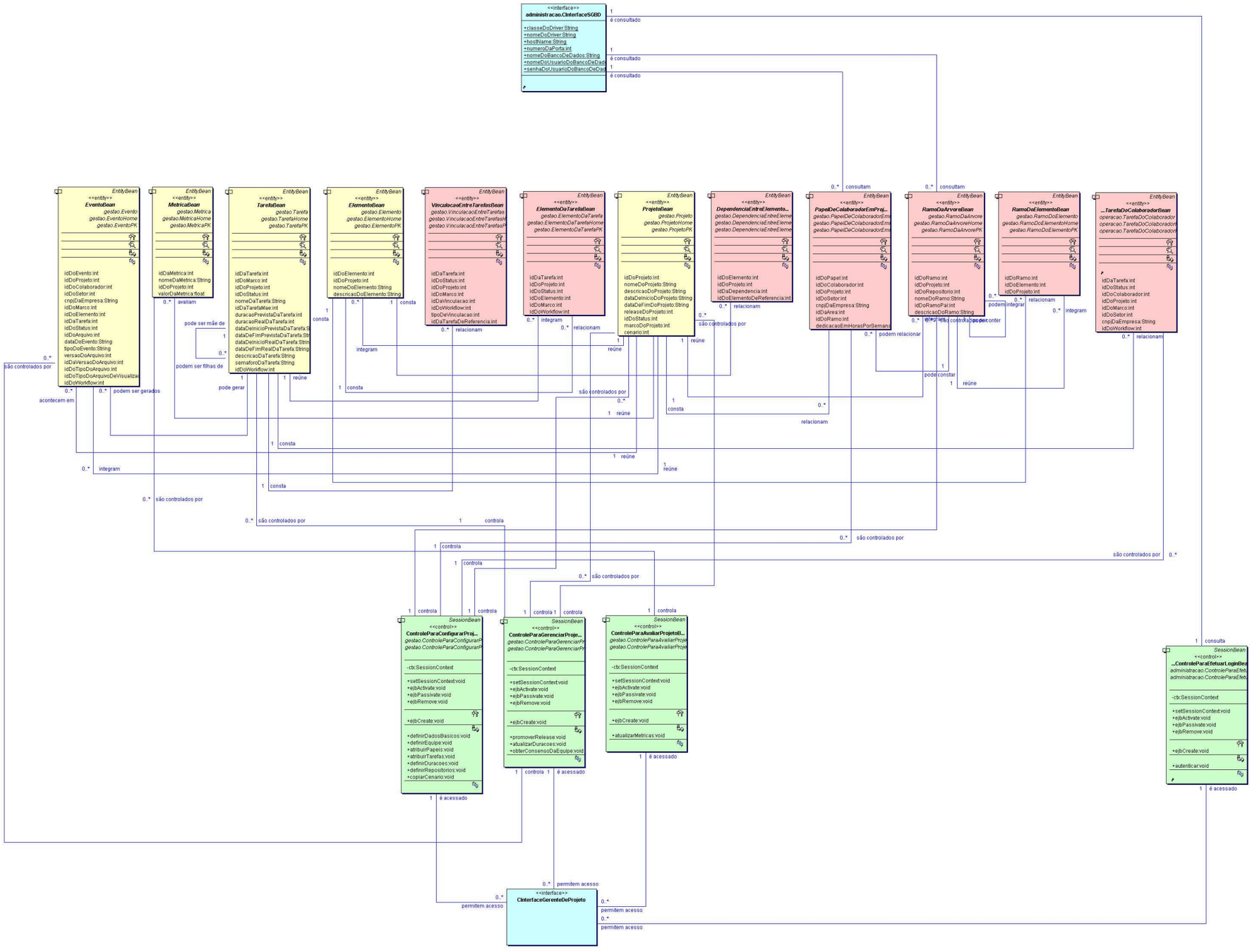
DLU | Housing | Back-up supply | Power supply | Overvoltage protection | MDF-DG | Heater | Painel AC-DC | Splice cassette | Fan | Sunroof | Equipo A | Equipo B | Equipo C

Pronto NUM

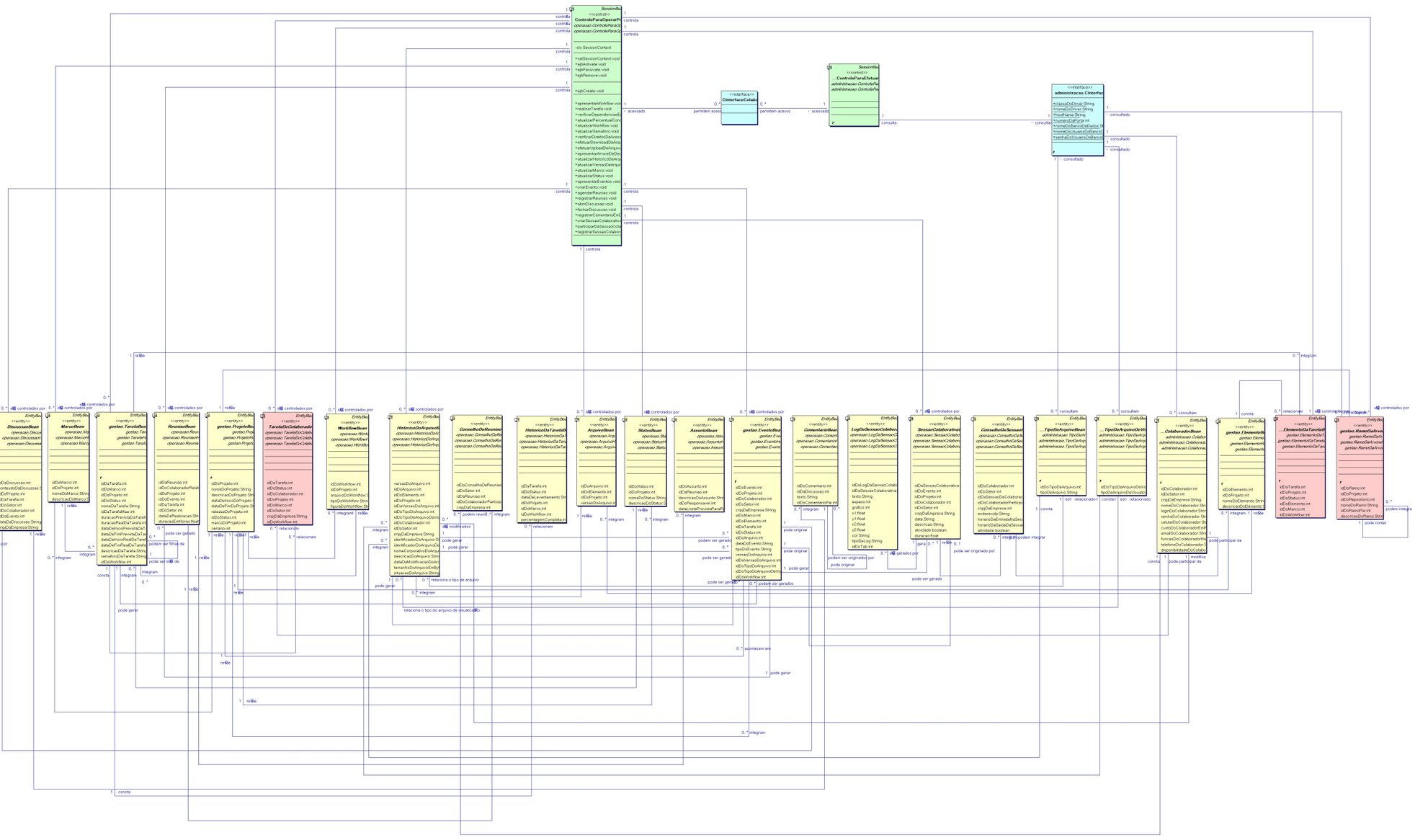
APÊNDICE 8 - ESTUDO DE RELACIONAMENTO ENTRE CLASSES DO MODULO DE ADMINISTRAÇÃO



APÊNDICE 9 - ESTUDO DE RELACIONAMENTO ENTRE CLASSES DO MODULO DE GESTÃO



APÊNDICE 10 - ESTUDO DE RELACIONAMENTO ENTRE CLASSES DO MODULO DE OPERAÇÃO



GLOSSÁRIO

.NET Plataforma proprietária da empresa Microsoft destinada ao desenvolvimento *Web* e de aplicativos distribuídos.

applet Um programa escrito em linguagem Java que pode ser distribuído como um anexo a um documento na internet e executado por navegadores habilitados para Java, talis como HotJava da empresa Sun Microsystems ou Netscape Navigator versão 2.0 ou posterior.

backup Uma cópia parcial ou completa dos arquivos armazenados nos discos rígidos de um computador, a qual é realizada periodicamente e mantida em meio magnético removível.

C Uma linguagem de programação concebida por Dennis Ritchie nos laboratórios da AT&T Bell em 1972 para programação de sistemas no PDP-11 e imediatamente utilizada para reimplementar o sistema operacional UNIX.

C++ Uma linguagem de programação orientada a objetos derivada da linguagem C.

chat Qualquer sistema que permita um número de usuários conectados a conduzirem uma conversação via teclado e em tempo real, tanto através do *login* de todos os usuários no mesmo computador, quanto através de uma rede de computadores.

check-in Processo de *upload* de arquivos em sistemas do tipo PDM.

check-out Processo de *download* de arquivos em sistemas do tipo PDM.

CSCW Ferramentas de software e tecnologia que apóiam grupos de pessoas que trabalham juntas num projeto, geralmente em locais distintos.

deployment Processo de instalação para uso de um aplicativo *web* num servidor do tipo "contêiner *web*".

display Monitor de vídeo.

download Transferência de dados ou (especificamente código) de um computador para outro. A distinção entre *download* e *upload* é obscura, porém *download* geralmente se refere a transferências de um sistema maior (servidor) para um sistema menor ("cliente").

EJB (*Enterprise JavaBeans)* Componentes *JavaBeans*utilizados na especificação

J2EE.

e-mail Mensagens automaticamente passadas de um computador para outro, geralmente através de redes de computadores e/ou via modems e linhas telefônicas.

empowerment Atribuição de poder, especialmente poder legal ou autoridade oficial.

FTP Um protocolo cliente-servidor que permite a um usuário transferir arquivos de/para um computador através de uma rede TCP/IP.

GIF Um padrão para imagems digitalizadas comprimidas com o algoritmo LZW, definido em 1987 pela CompuServe (CIS).

go-to Um comando encontrado em várias lingugens de programação de alto nível (e.g. FORTRAN, COBOL, BASIC, C) que causam um salto incondicional ou transferência de controle de um ponto a outro num programa.

groupware Ver CSCW.

HTML Um formato de documento em Hipertexto utilizado na internet. "Hipertexto" é um termo cunhado por Ted Nelson por volta de 1965 para uma coleção de documentos (ou "nós") contendo referências cruzadas as quais, com a ajuda de um navegador interativo, permitem que o usuário se movimente mais facilmente de um documento para outro.

HTTP Protocolo TCP/IP cliente-servidor utilizado na internet para a troca de documentos HTML.

intranet Uma rede de computadores que fornece serviços dentro de uma organização similares àqueles fornecidos pela internet, porém não necessariamente conectado à mesma.

J2EE Especificação de plataforma Java multi-camadas orientada a servidores desenvolvida pela empresa Sun Microsystems para aplicações corporativas. A base da especificação J2EE são os EJBs.

J2EE Patterns Padronização de componentes funcionais utilizada para aplicativos desenvolvidos de acordo com a especificação J2EE.

Java Uma linguagem de programação orientada a objeto criada pela empresa Sun Microsystems.

JavaBeans Uma arquitetura de componentes para a linguagem de programação Java, desenvolvida inicialmente pela Sun Microsystems, porém agora disponibilizada por vários outros fabricantes. JavaBeans são também chamados simplesmente de "beans". JavaBeans permitem aos desenvolvedores criarem componente de *software* reutilizáveis, que podem ser reunidos num aplicativo através de ferramentas visuais tais como Sybase PowerJ, Borland JBuilder, IBM Visual Age for Java, SunSoft Java Workshop e Symantec Visual Cafe.

JavaScript A linguagem multi-plataforma simples baseada em *scripts* para *Web* desenvolvida pela empresa Netscape.

JDBC Parte do *kit* de desenvolvimento Java que define a programação de interfaces para aplicações em Java, destinada ao acesso padrão SQL a bancos de dados.

JPG (ou "JPEG") O nome original de um comitê que projetou o algoritmo padrão de compressão de imagens.

JSP Uma especificação para estender a funcionalidade de servlets Java a fim de gerar páginas *Web* dinâmicas num servidor *Web*.

lead time Tempo total necessário para completar uma unidade de um produto ou serviço.

log Coleção de registros.

logon (or "log in", "log on", "logon") Abertura de uma sessão num sistema, geralmente através do fornecimento de um nome de usuário e uma senha como meio de autenticação.

markup Um método para adicionar informações a um texto indicando instruções, comentários ou outro tipo de anotação.

milestone Evento significativo ocorrido durante a execução de um projeto.

MPEG-2 Uma variação do algoritmo de compressão de vídeo e áudio MPEG, otimizado para transmissão de vídeo em TV digital e DVD.

off-line Não diretamente conectado a um outro computador ou com conexão suspensa.

on-line Conectado a um outro computador.

parsing Um algoritmo ou programa para determinar a estrutura sintática de uma seqüência de palavras ou símbolos numa linguagem.

PHP Uma linguagem multi-plataforma baseada em scripts embutida em documentos HTML utilizada para a criação de páginas *Web* dinâmicas. Trata-se de *software* livre.

plug-in Um arquivo que contém dados utilizados para alterar, melhorar ou estender a operação de um aplicativo principal.

PMXML Padrão XML específico para a descrição de dados de projeto.

pull-down (ou *drop-down list*) Um componente de interface gráfica que permite ao usuário escolher um (ou às vezes mais de um) item de uma lista.

redesign Correção de projeto já iniciado.

release Uma versão de um projeto que pode ser publicada (em contraste a outra que esteja em desenvolvimento).

scriplet Porção de código Java que insere lógica de programação em JSPs.

script Um programa escrito em linguagem baseada em *scripts*. Linguagens baseadas em *scripts* necessitam de poucos comandos, não dispõem de recursos para tratamento de estruturas complexas de dados e os programas que se utilizam delas devem ser "interpretados". Isto significa que os *scripts* devem interagir com outros programas ou conjuntos de funções fornecidas por um interpretador (e.g. funções fornecidas pelo sistema operacional UNIX).

servlets Um programa Java que funciona como parte de uma rede de serviços, tipicamente um servidor HTTP e responde a solicitações de clientes.

SQL Uma linguagem padrão para a criação, atualização e busca em sistemas de bancos de dados relacionais.

staffing Alocação de recursos humanos em projetos.

time-to-market Tempo transcorrido desde o conceito inicial de um produto até a sua introdução no mercado.

top-down Técnica de projeto de *software* que objetiva descrevê-lo funcionalmente num nível muito alto, depois particioná-lo repetidamente em níveis mais detalhados, até que os detalhes sejam suficientes para permitir a sua codificação.

upload Transferência de programas ou dados através de uma linha de comunicação digital a partir de um sistema menor ou "cliente" para um sistema maior ou

"servidor".

VRML Uma especificação para o projeto e implementação de uma linguagem independente de plataforma para a descrição de imagens de realidade virtual.

web (ou *World-Wide Web*) > "A *Web*" é o ambiente aberto da internet. "Uma *web*" é uma parte dela ou algum sítio específico.

whiteboard Painel eletrônico que aceita marcações indicativas de comentários.

workflow Conjunto de relacionamentos entre as atividades de um projeto, do começo ao fim. As atividades são relacionadas por diferentes tipos de vínculos entre si.

wrappers Elementos de síntese.

XML Uma iniciativa do comitê internacional W3C para definir um dialeto de SGML (*Standard Generalized Markup Language*) "extremamente simples", adequado ao uso na Web. Originalmente projetado para suprir as necessidades de publicações em larga escala, o XML desempenha um papel cada vez mais importante na troca de uma grande variedade de dados na *Web*.